APPLICATION OF HOLOGRAPHIC INTERFEROMETRY TO SUPERSONIC FLOW VISUALIZATION OF A REAR-FACING STEP AND APPLICATION OF FAST FOURIER TRANSFORM METHODS TO CURRENT DATA REDUCTION TECHNIQUES

Harland Wilber Jones

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

APPLICATION OF HOLOGRAPHIC INTERFEROMETRY TO SUPERSONIC FLOW VISUALIZATION OF A REAR-FACING STEP AND APPLICATION OF FAST FOURIER TRANSFORM METHODS TO CURRENT DATA REDUCTION TECHNIQUES

by

Harland Wilber Jones, Jr.

Thesis Advisor: D. J. Collins

March 1974

Application of Holographic Interferometry to Supersonic Flow Visualization of a Rear-Facing Step and Application of Fast Fourier Transform Methods to Current Data Reduction Techniques

by

Harland Wilber Jones, Jr.
Lieutenant, United States Navy
B.S., United States Naval Academy, 1968

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
March 1974

ABSTRACT

The successful application of holographic interferometry to the study of three-dimensional density fields around bodies in wind tunnel experiments has been reported in the literature along with the associated mathematical reduction processes for the basic interferometric equation.

The present report has extended the application of holography as a flow visualization technique by investigating Mach 2.8 flow over a rear-facing 1/8 in. step. Two single-exposed holograms of the model and two double-exposed frozen fringe holographic interferograms of the flow were created.

Fast Fourier Transform (FFT) methods have been reported as significantly reducing computational time to obtain discrete and inverse discrete Fourier transforms. Two computer programs were designed to apply FFT methods to the Fourier transform approach to the basic interferometric equation. These programs were used to investigate an alternate way to calculate the value of the inside integral. Total agreement was not found, and further analysis as to the accuracy of each method is needed.

TABLE OF CONTENTS

LIST OF FIGURES

TABLE OF SYMBOLS

Symbol	Meaning
DFT	Discrete Fourier transform
DR	Interval between frequency plane values, used as a weighting function for determining the convolution
F(A)	Operator meaning "to take the discrete Fourier transform of 'A'"
$F^{-1}(A)$	Operator meaning "to take the inverse discrete Fourier transform of 'A'"
FFT	Fast Fourier transform
IDFT	Inverse discrete Fourier transform
L	Length
M	Mach number
mW	Milliwatt
n, n_o	Index of refraction, reference index of refraction
NP	Number of points of frequency plane values
Q	Number of frequency plane values in the finite sample
r	Radius
R	Radius of convolution vector
RMAX	Maximum radius of original finite frequency plane values
S_i	The i^{th} region of integration
dS_i	Distance along the i^{th} ray traversing the test section
t	Step size of rear-facing step
U	Local velocity
$\mathring{A}$	Angstrom degree, equal to 10^{-8} cm.
γ	Power of 2, i.e. 2^{γ}
(C1*C2)	"*" Operator meaning "C1 is convoluted with C2"

$\frac{\partial(a)}{\partial b}$	Operator meaning the "partial derivative of 'a' with respect to 'b'"
θ	Angular measurement in polar coordinates
ν	Coefficient of viscosity for fluid involved in the flow under consideration
φ	View angle
φ_i	Known path length function; frequency plane values
ρ	Distance from origin in polar coordinates
ρ, ρ_o	Density, reference density

ACKNOWLEDGMENTS

The writer wishes to gratefully acknowledge Dr. D. J. Collins for his most valuable guidance and assistance throughout the course of this investigation; the technical staff of the Department of Aeronautics under R. Besel and T. Dunton, particularly N. Leckenby for his invaluable assistance in construction and operation of the experimental equipment and G. Middleton for his fabrication of the experimental model; Deborah Janov for her considerable effort in the final preparation and typing of this report; and my wife and family for their noble patience and encouragement.

I. INTRODUCTION

This thesis contains both an experimental section and a theoretical section. The experimental portion involved the construction of a modified Z-type Schlieren system to conduct holographic investigation of a rear-facing step in supersonic flow. The theoretical portion involved investigating current data reduction techniques for determining the three-dimensional distribution of a field by applying fast Fourier transform methods to solve the basic interferometric equation. Two original computer programs were devised in an attempt to reduce computer computational time required for interferometric data reduction.

II. INTRODUCTION TO HOLOGRAPHY

Holography is a three-dimensional photography-like process which involves the recording of both phase and amplitude information of light waves diffracted by an object illuminated with coherent light. Originally proposed by Gabor [1,2] in 1948, holography awaited the development of the laser around 1960 to become an explosive field of its own. The laser contributed a basically monochromatic, coherent light source. The property of coherence, or the capability of interference with itself, enables the recording of both phase and amplitude information.

Photography records the intensity of illumination, which is the time average of the square of the amplitude of the light field at any point. In holography, a coherent light beam is separated into a reference beam and an object (or scene) beam. The reference beam is directed to the holographic plane via an optics system while the object beam reaches the same point after being diffracted by a subject of interest. The emulsion on the holographic plate records the intensities of the illuminating beams and the interference between the two beams. The interference component contains the phase information which permits reconstruction of an an image possessing full three-dimensionality and parallax properties. No attempt will be made to present the full mathematical and historical development of holography which is amply done in many excellent texts. [3]

Among its many engineering applications, holography is well suited for flow visualization. Recent successful applications of holography at the Naval Postgraduate School include the study of a free jet and the determination of the density field about a cone tilted in supersonic flow in a conventional wind tunnel. The use of double-pulsed holography is

especially helpful as a tool for measurement in high velocity flows near vibrating equipment such as free jets and blow-down wind tunnels. A Q-switch ruby laser is used as a high power energy source emitting a coherent light pulse on the order of several joules in an extremely small time span, on the order of twenty nanoseconds. Vibrations of the object or of the optical system components, which must remain stable within wave lengths of light during exposure of the hologram, are of negligible importance due to the very short duration of the exposure time.

A double-pulsed (or double-exposed) approach is advantageous in flow visualization. In the Mach-Zender interferometer, comparison of the reference and object beams are made at the same time. With holography it is possible to separate the simultaneous comparison in time. Two holograms, or exposures, are made on the same holographic plate without intermediate development. The first exposure is made with no flow and the second with flow. The individual holograms interfere with each other, creating an interferogram of the flow. Essentially, this interferogram contains only the differences in optical path lengths between the two exposures. Therefore, any imperfections in optical components, (which are identical for both exposures), are subtracted out, thereby eliminating the need for high quality optics [4].

Another method of taking finite fringe holograms is the double-exposure option of the dual hologram method, which involves using two holographic plates. One plate is used for each exposure. A dual hologram holder positions each plate in the same location for its exposure. The improvement comes from the versatility of almost an infinite set of reconstruction possibilities from one set of plates. A dual hologram holder has recently been constructed at the Naval Postgraduate School to demonstrate its application. This topic was the subject of AIAA Paper 73-210 [5].

III. INVESTIGATION OF A REAR-FACING STEP

Holographic investigation of a rear-facing step in supersonic flow was selected for two main reasons. First, extensive theoretical and experimental investigations of a rear-facing step in two-dimensional supersonic flow are available. This provides ample information for verifying results from this experiment. Secondly, previous experimental results do not completely agree with theoretical descriptions. A source of conflict is the lip shock's influence on the rest of the flow as it pertains to the reattachment point and to the tail shock. With this in mind, it was hoped to provide a hologram which could be used to accurately reproduce the flow field.

The classical flow model for supersonic base flow, (rear-facing step flow), was proposed by Chapman [6] in 1957 and independently by Korst [7] for turbulent flow, and is commonly known as the Chapman-Korst flow model. Refer to Figure 1 for a sketch of the Chapman-Korst flow model. Roache [8] presented a literature search and a review of discrepancies with this model in his doctoral thesis for the University of Michigan in 1968. The main, unresolved discrepancies in the Chapman-Korst flow model were: that the experimental pressure at recompression was not equal to the theoretical pressure of Region Four (see Fig. 1); and that the isentropic turn from Regions One to Two (see Fig. 1) and the constant pressure assumptions were invalidated by the presence of the lip shock (see Fig. 2) at higher Mach numbers. [9]

The opportunity to provide some accurate data on these unresolved areas of interest appeared to be a worthwhile application of holography.

IV. EXPERIMENTAL MODEL

The actual rear-facing step model used was limited in size, with the maximum step size being one-eighth inch. Due to the small step size, the boundary layer thickness (δ) was compared with the step size (t) to determine if distortions from the boundary layer would invalidate experimental results. From Schlicting [10], the applicable equations are: $\delta = 5\ (\nu\ L\ /\ U)^{\frac{1}{2}}$ for laminar flow; and $\delta\ /\ L = .37\ (\ U\ L/\ \nu)^{-1/5}$ for the boundary layer thickness at the trailing edge of a flat plate at zero incidence in parallel turbulent flow. The $\delta\ /\ t$ ratio results were .0538 for laminar flow and .218 for turbulent flow. It was felt that the actual flow situation was laminar and that this step size would still offer meaningful results.

A photograph of the model is included as Figure 3. Refer to Figure 4 for a sketch showing the dimensions of the model.

V. DESCRIPTION OF APPARATUS

Figure 5 shows a complete sketch of the experimental setup. A modified Z-type Schlieren system, it was constructed specifically for this thesis. A description of the integrated setup will be followed by listing of important supportive equipment.

A high energy pulse was discharged from a Q-switch ruby laser and passed through a diffusing lens to a collimating mirror in order to expand the light beam to sufficiently illuminate the test section. The collimated beam reflected off a flat mirror towards the test section. Just prior to the test section, the beam was divided by a beam splitter mounted on a tripod. The object (scene) beam transmitted through the beam splitter, through the test section, and reflected off another flat mirror towards a converging mirror. The object beam was then reflected towards the holographic plane which was positioned in the diverging part of the beam where the test object was in focus. The part of the collimated beam reflected by the beam splitter, the reference beam, reflected off three flat mirrors and also illuminated the holographic plane.

Other equipment included an air supply for the tunnel provided by an air compressor; a power supply and cooling system for the laser; a 15 mW continuous wave Helium-Neon laser used for alignment of the Q-switched ruby laser with the optics system; an autocollimator for internal alignment of the Q-switch laser; a movable platform for the laser so it can be elevated to the proper height for the optics system; a table for the hologram holder; and mountings and supports for all optic components. Figures 6-9 are pictures showing the experimental setup from four different views.

Safety equipment included eye protectors and sound attenuators.

VI. EXPERIMENTAL PROCEDURE

Appendix A contains a detailed description of the entire experimental procedure from alignment of system components to exposure of various types of holograms.

There are two holographic methods that can be applied to flow visualization in wind tunnels. The first method is commonly referred to as the real fringe method, and the second as the frozen fringe or double-exposed method.

Real fringe methods require exposure of the holographic plate with no flow in the wind tunnel and either development of the hologram in place or precise repositioning of the developed plate. As flow is increased in the wind tunnel, an evolution of real fringe patterns could be observed through the hologram. This method was not used in this experiment due to the unavailability of in-place developing equipment or a dual-hologram holder.

Double-exposed or frozen fringe methods require two exposures of the same holographic plate. The first exposure is taken with no flow in the wind tunnel, and the second with flow. The interference between the two exposures results in a double-pulsed (exposed) interferogram. In this experiment both exposures were made on the same plate with the second exposure occurring while Mach 2.8 flow existed in the test section.

VII. FILM AND DEVELOPMENT PROCEDURE

There are essentially two sources of holographic films: Eastman Kodak Company and Agfa-Gevaert. Agfa-Gevaert films were used exclusively in this experiment because their 75 series film is designed for maximum sensitivity around the emission spectrum of ruby lasers (6943 Å). Due to the short duration of the exposure time of the Q-switch ruby laser, about 10-20 nanoseconds, 10E75 film was used because 8E75 film turned out to be too slow in recording information.

Development of holograms can include various features such as bleaching, prehardening, or varying development time. The various alternatives are examined in Ref. 11. For this experiment the following development procedure was used:

1. Five minutes in Kodak D-19 Developer
2. Thirty seconds in an acetic acid stop bath of standard dilution.
3. Five minutes in a standard fixer.
4. Five minutes of washing followed by immersion in a wetting agent (Kodak Photoflo) prior to drying.

VIII. RECONSTRUCTION TECHNIQUE, EQUIPMENT, AND RESULTS

A 15 mW continuous wave Helium-Neon laser, previously used for alignment, was the coherent light source used for reconstruction. A collimating lens was attached to this laser in order to expand the reconstruction beam to approximately two inches. Other reconstruction equipment included a hologram holder, type 52P Polaroid film, a graphic view camera, a tripod, and a black drape.

The two-inch diameter collimated beam incident to the hologram created two images. Whether projected images are real or virtual is discussed in Chapter 3 of Ref. 3 for the various combinations of converging, diverging, and collimated reference and scene beams. In this experiment, approximate symmetry about the vertical axis of the holographic plate was attempted. This resulted in the real image being projected as a converging wave front and in the virtual image being projected as a diverging wave front.

In order to focus the real image, the lens was removed from a camera and the image was projected on the ground glass backing of the camera. Pictures of these projections were taken at exposure times which varied with the intensity of the real image beam. Normal exposure times were approximately one second.

Figure 10 is a sketch of the reconstruction setup and Figures 11-14 are examples of photographs of the projections of the real image of holograms taken with this experimental setup.

Figures 13 and 14 represent the final experimental attempts to create a frozen fringe hologram of sufficient quality to warrant data reduction. The edges of the experimental model show that both holograms are slightly out of focus. These holograms were, however, sufficient to demonstrate

the applicability of holography to flow visualization of a rear-facing step in supersonic flow. Additionally, the shadowgraph resulting on the developed plate showed the basic features of the Chapman-Korst flow model. Sufficient information was not obtained to analyze any of previously mentioned discrepancies of that model.

The remaining effort of this thesis focused on the possibilities of improving current data reduction methods.

IX. CURRENT DATA REDUCTION TECHNIQUES

There are currently several approaches for reducing an asymmetric density field. One method involves a series expansion of a complete set of orthogonal polynomials which are invariant to a rotation of coordinates [12]. Matulka, [13] used this method and created a computer program which took approximately two hours of computer time to reduce the data for the asymmetric case.

A second approach uses Fourier transform techniques suggested by Rowley [14] and further developed by Junginger and van Haeringen [15]. This approach is thoroughly discussed by Sweeney [16] in his doctoral thesis for the University of Michigan in 1972.

Closely following Sweeney's development and nomenclature, the fundamental integral equation to be solved in three-dimensional interferometry is

$$\int_{S_i} f(x,y,z)\, dS_i = \varphi_i \qquad (9\text{-}1)$$

where $f(x,y,z)=n(x,y,z)-n_o$; $f(x,y,z)$ is the refractive index relative to a known value n_o, S_i is the distance along the i^{th} ray traversing the test section, φ_i is the known path length function, and the integral is evaluated between boundaries imposed by experimental configuration or by prior knowledge regarding $f(x,y,z)$. The objective is to reconstruct a real scalar function, the relative refractive index field, from the scalar values of line integrals through the field. When the gradient of the refractive index variation is small enough to produce negligible ray curvature, eqn. (9-1) alone is the basic interferometric equation to be solved. The line integrals are then evaluated along straight, not curved, lines.

The presentation of the problem is simplified by the suppression of one of the dimensions. Solutions for the two-dimensional case can be layered to reproduce the three-dimensional distribution, thereby permitting simplification without loss of information.

The use of polar coordinates on the two-dimensional equivalent of eqn. (9-1) yields

$$\int f(x,y)\, dS = \varphi(\rho,\theta) \tag{9-2}$$

where ρ varies to cover the domain of $f(x,y)$ and θ varies between $\pm \pi/2$. Eqn. (9-2) then becomes the fundamental integral equation to be solved.

If the refractive index field is radially symmetric or if it is independent of one of the coordinates, then the solution of eqn. (9-2) is known and has been extensively used in interferometric studies. For $f(x,y)=f(y)$ only, eqn. (9-2) becomes

$$\varphi(y) = \int_{-L/2}^{L/2} n(y)\, dx = L\, n(y) \tag{9-3}$$

and the solution is $n(y) = \varphi(y)\, L$.

For the radially symmetric case, only one field of view is necessary and eqn. (9-2) becomes

$$\varphi(x) = 2 \int_{r=x}^{\infty} f(r)\, dy = 2 \int_{x}^{\infty} \frac{f(r)\, dr}{(r^2-x^2)^{\frac{1}{2}}} \tag{9-4}$$

which is a form of Abel's integral equation.

The solution of eqn. (9-4) may be obtained by converting the integral into a convolution integral by using appropriate transformations. The solution, $f(r)$, then becomes

$$f(r) = \frac{-1}{\pi} \int_{r}^{\infty} (x^2-r^2)^{\frac{1}{2}} \frac{d}{dx}\left[\frac{\frac{\partial \varphi}{\partial x}}{x}\right] dx \tag{9-5}$$

Sweeney uses the definitions of Fourier transforms, a few fundamental transform pairs, the procedure for inverting the Fourier transformed

equivalent of eqn. (9-2) as outlined by Junginger and van Haeringen [15], and an introduction of polar coordinates to the transformed equation to yield

$$f(r,\varphi) = \frac{1}{2\pi^2} \int_{-\pi/2}^{\pi/2} d\theta \left[\frac{\partial \varphi(z,\theta)}{\partial z} * \frac{1}{z} \right] \tag{9-6}$$

where "*" denotes the convolution. Expanding the convolution integral and replacing z with $r \sin(\varphi-\theta)$, the final result for $f(r,\varphi)$ is:

$$f(r,\varphi) = \frac{1}{2\pi^2} \int_{-\pi/2}^{\pi/2} d\theta \int_{-\infty}^{+\infty} \frac{\partial \varphi(\rho,\theta) / \partial \varphi}{r \sin(\varphi-\theta) - \rho} \partial \rho \tag{9-7}$$

This is the basic relationship between the pathlength function and the refractive index field.

Referring to eqn. (9-6), the integral over θ is a linear summation of the contributions for each direction of view. The contribution from each value of θ is equal to the convolution of the derivative of the path length function and, (1/z) with (1/z) evaluated at $z = r \sin(\varphi-\theta)$

When $\rho = r \sin(\varphi-\theta)$, the integrand of eqn. (9-7) becomes singular. This presents no problem since the contributions to the integral near the origin tend to cancel for an odd function (like 1/z). Thus the integral remains finite.

The integration over ρ in eqn. (9-7) is a Hilbert transform of $\varphi(\rho,\theta)$ and the properties of Hilbert transforms along with the use of discrete data, guarantee the boundness of eqn. (9-7). Both this form of the integration over ρ and Van Houten's [17] modification to it will be called the inside integral in the next section.

A finite number of values of the frequency plane must be determined. One method for their determination uses linear interpolation from adjacent radial lines of a superimposed grid system. Success of this method depends on the accuracy of the interpolation. If the pathlength data is

taken for a sufficiently large number of angular orientations, linear interpolation may be used.

These linearly interpolated frequency plane values ($\varphi(\rho,\theta)$) must be convoluted with 1/z, i.e.

$$[\varphi(\rho,\theta) * 1/z] \tag{9-8}$$

with "*" meaning convolution.

Using the following notation of F for obtaining the discrete Fourier transform (DFT) and F^{-1} for obtaining the inverse DFT or IDFT, the remaining steps required to perform the integration over ρ are summarized as follows:

The DFT of the convolution is obtained

$$F[\varphi(\rho,\theta) * 1/z] \tag{9-9}$$

followed by the differentiation of this function

$$d(F[\varphi(\rho,\theta) * 1/z]) \tag{9-10}$$

Finally the value for the integration over ρ is obtained by taking the IDFT of the preceeding function.

$$F^{-1}\,[d(F[\varphi(\rho,\theta) * 1/z])] \tag{9-11}$$

Van Houten [17] used the linear interpolation approach to obtain the frequency plane values by a computer program designed to calculate the three-dimensional density field from data obtained from holographic interferometry. The frequency plane values for the density field are represented by expanded fringe shift information calculated from the fringe number of specific points on the interferogram. Van Houten modified eqn. (9-7), the basic interferometric equation to the following:

$$f(x_o,y_o) = \frac{-1}{2\pi^2} \int_0^{\pi} \left[\int_0^{3RMAX} \frac{g(\rho_o+\rho,\varphi) + g(\rho_o-\rho,\varphi) - 2g(\rho_o,\varphi)}{2}\, d\rho + \frac{2g(\rho_o,\rho)}{3RMAX} \right] d\varphi \tag{9-12}$$

with $g(\rho,\varphi)$ being the expanded fringe shift information and the integral inside the brackets referred to as the inside integral instead of the integration over ρ.

Van Houten's computer program uses Cote's sixth order method [18] for integrating the first term of the inside integral. He applied a correcting term for the contribution of the integral from 3RMAX to infinity. The sum of these two terms is called the inside integral. If NP is the number of data points for each view, and if K is the number of views, the inside integral must be evaluated (NP · K) times with Van Houten's computer program. Additionally, for each of the NP^2 grid intersections, K interpolations for $G(\rho_o,\varphi_k)$ are required. The subsection of Van Houten's computer program which calculates the value of the inside integral is included in Appendix B. The next section will investigate alternatives for this part of Van Houten's computer program.

X. APPLICATION OF THE FAST FOURIER TRANSFORM

Fast Fourier transform (FFT) methods are simply a more efficient means of obtaining the discrete Fourier transform (DFT). Their increased efficiency is due to the significant reduction in the number of complex mathematical operations required. For a function of N points, normal calculation of the DFT requires nearly N^2 operations while the FFT radix-2 method reduces this to $(N/2)\log_2(N)$ complex multiplications, $(N/2)\log_2(N)$ complex additions, and $(N/2)\log_2(N)$ complex subtractions. As can be seen from Figure 15, for N=1024, this results in a computational savings of more than 200 to 1. [19] Higher radix FFT methods further reduce the number of operations required.

Operations usually associated with FFT applications include: (1) computing a spectrogram; (2) computing the convolution of two time series; and (3) computing the correlation of two time series. The second of these operations has been shown to have direct application to the calculation of the value of the inside integral for determining the three-dimensional density field. Reference 20 gives a thorough discussion of the FFT and the required steps for convolution and correlation. These detailed procedures for computing the convolution are included in Appendix C for the following combination of functions: (1) two finite length functions; (2) a finite and an infinite length function using the select-savings method; and (3) a finite and an infinite length function using the overlap-add method.

Two original Fortran programs were devised to apply FFT methods to the calculation of the values of the inside integral. Specifically, alternatives to Mode One operation of Van Houten's computer program for the

asymmetric Gaussian case of $f(z) = (1-y^2)\exp(-2x^2)$ for 65 points from $-1.5 \leq R \leq 1.5$ were investigated. Mode One operation of Van Houten's program requires refractive index information supplied by SUBROUTINE INPUT. Using a finite sample of 65 points with an interval between samples of .046875, Van Houten's program required 1.3388 seconds of computer time just to calculate the value of the inside integral. This time was obtained by using computer library subroutines SETIME and GETIME placed just prior to and immediately after the subsection of Van Houten's computer program which calculates the value of the inside integral.

Van Houten's program also uses a SUBROUTINE COEFF to calculate coefficients used in various parts of the program. MODE1, a modification of Van Houten's program which eliminated the need for SUBROUTINE COEFF by returning its calculations to the main program, is included in Appendix D. This modification includes the essentials of Van Houten's thesis with the following corrections: C1 = .292857 instead of .2292857; several divisions by 59.296 were corrected to 57.296; and all plotting directions were deleted due to the storage space limitation of the IBM 360 on-line time sharing capability. This modification followed the work done by Everett [21].

Both FFT method programs used a finite sample of the expanded fringe shift information obtained from Van Houten's computer program. His method for obtaining these frequency plane values followed the linear interpolation approach as described by Sweeney. Sixty-five finite sample points were obtained and they had to be convoluted with the infinite function 1/z. JONES1 numerically convoluted these functions while JONES2 used the FFT select-savings method to obtain the convolution. Once the convolution was obtained, both programs followed the procedures described in the preceding section, except that FFT methods were used to obtain all

DFT's and IDFT's. The two separate approaches for calculating the convolution served as an internal check that the correct convolution was being obtained. The complete programs are included in Appendix F. along with a listing of the expanded fringe shift information obtained from Van Houten's program and used as frequency plane values for both programs.

The numerical methods used in JONES1 required multiplying the final result by the appropriate weight function, the interval between sampled points of the expanded fringe shift information. JONES2 used the FFT select-savings method to calculate the convolution. The dimension of the 1/R vector was chosen to be 512 following Brigham's [20] recommendation as the optimal choice for values of Q between 50 and 99. His table of recommendations for the size of arrays needed for FFT convolution is included in Figure 16. The sample interval for both of these FFT programs was selected to agree with the interval used in Van Houten's program, namely .046875. For N = 512, and DR being defined as 2RMAX/N, an R of 12.0 was required. Using FFT methods to obtain the Fourier coefficients of the convolution also required a weighting function. In the complex multiplication of the normalized Fourier coefficients of the expanded fringe shift information and the 1/R vector, the resulting Fourier coefficients incorporated an extra division by N. Instead of a weighting function of DR then, 2R was chosen as the appropriate weighting function. When the IDFT of the Fourier coefficients was obtained, the values of the convolution by this FFT method were in exact agreement with the convolution obtained by numerical methods.

The differentiation technique used in JONES1 and JONES2 required an increment between steps of differentiation that was equal to the sample data interval. Thus, the same DR was used for this process.

All FFT calculations in both programs utilized SUBROUTINE CFFT2, an algorithm for computing the mixed radix fast Fourier transform [22]. The complete single precision version of CFFT2 is included in Appendix E. Chosen for its versatility, it contains a detailed and straight-forward usage description at the beginning of the subroutine. Several known Fourier transform pairs were investigated to ensure that CFFT2 was being used correctly. Correct DFT's and IDFT's were easily obtained.

As was previously mentioned, FFT methods involve complex mathematical operations. When using only real functions, as was the case with the expanded fringe shift samples and the 1/z function, the frequency plane values could be placed in the real coefficient array of CFFT2 and the other function could be placed in the imaginary coefficient array of the same call to CFFT2. This would result in the significant reduction of time and storage space required to calculate DFT's and IDFT's. This increased efficiency was <u>not</u> utilized in either JONES1 or in JONES2. Instead, both programs are quite inefficient, with redundant storage used for recording and checking each step in the entire process of calculating the value of the inside integral. If one only desired a print-out of the inside integral values, these programs could easily be reduced to four matrices, and possibly to two matrices since both functions were real in this case. Both programs, designed as possible alternatives for Van Houten's subsection that calculated the value of the inside integral, were only written to replace inside integral calculations for Mode One operation. Hence, in addition to condensing these programs into efficient packages, investigation of all other modes of operation is required before a complete integration of FFT methods into Van Houten's program can be achieved.

XI. RESULTS AND RECOMMENDATIONS FOR FFT APPLICATION

The computer time required for just the calculation of the inside integral subsection of Van Houten's program for Mode One operation of the asymmetric Gaussian case took 1.3388 seconds. Although the increased efficiency of FFT methods are significant for large values of N, it was not known if FFT methods would reduce computational time required in this case where N was only 64 sample intervals. Nevertheless, if time had permitted integration of a condensed and efficient form of either JONES1 or JONES2 into Van Houten's program, it was felt that some reduction in computation time would have been achieved.

Both approaches to obtain the convolution, JONES1 using numerical methods and JONES2 using FFT methods, produced convolution values in exact agreement with each other and inside integral values within .002 per cent of each other. A listing of these results side by side to the inside integral values obtained from Van Houten's program is included in Appendix G. There was general agreement between the values of the inside integral as calculated by Van Houten's program compared with those obtained by FFT methods. Near a value of R equal to ± .8 the apparently significant percentage difference between the two methods was misleading since the magnitudes involved were small. However, for values of R near the discontinuity in the density, there was significant disagreement in the values of the inside integral. Further analysis as to the accuracy of the methods involved is required to resolve this problem.

Shifting the expanded fringe shift information from the middle of its array to the beginning of an array whose remaining points were filled with zeros, as suggested by the select-savings method, resulted in the

inside integral values corresponding to the $-1.5 \leq R \leq 1.5$ interval being stored in the output array beginning with the $(N/2 + 1)$ position. In this case, the appropriate values of the inside integral were stored from the 257^{th} through the 321^{st} positions of the output vector. According to the select-savings method (refer to Appendix C.2), the first "Q-2" points of the output vector are meaningless. Q is defined as the number of frequency plane samples used in the calculation of the convolution.

XII. SUMMARY OF RESULTS

An experimental setup was designed and constructed to conduct holographic investigation of a rear-facing step in supersonic flow. Although the double-exposed (frozen) holograms produced did not warrant further analysis of discrepancies of the Chapman-Korst flow model, they were of sufficient quality to demonstrate an application of holography to rear-facing step flow visualization near vibrating equipment such as the supersonic blowdown wind tunnel.

Theoretical investigations of current data reduction techniques for determining the three-dimensional density field from interferometric data led to investigation of the applicability of fast Fourier transform methods to reduce computational time required for determining the inside integration over ρ in the basic interferometric equation. Two programs were devised which calculated the value of the inside integral using FFT methods. Although the convolution was obtained differently for each program, numerically in JONES1 and with FFT's for JONES2, the convolution's were identical. Comparison of these values with those obtained from Van Houten's program lacked close agreement for the $\pm$ (1.07 - 1.31) intervals for R. The difference between these methods requires further analysis as to the accuracy of the methods. Further analysis is also needed to efficiently reduce JONES 1 and JONES2 and to integrate the programs into Van Houten's program for all of its modes of operation.

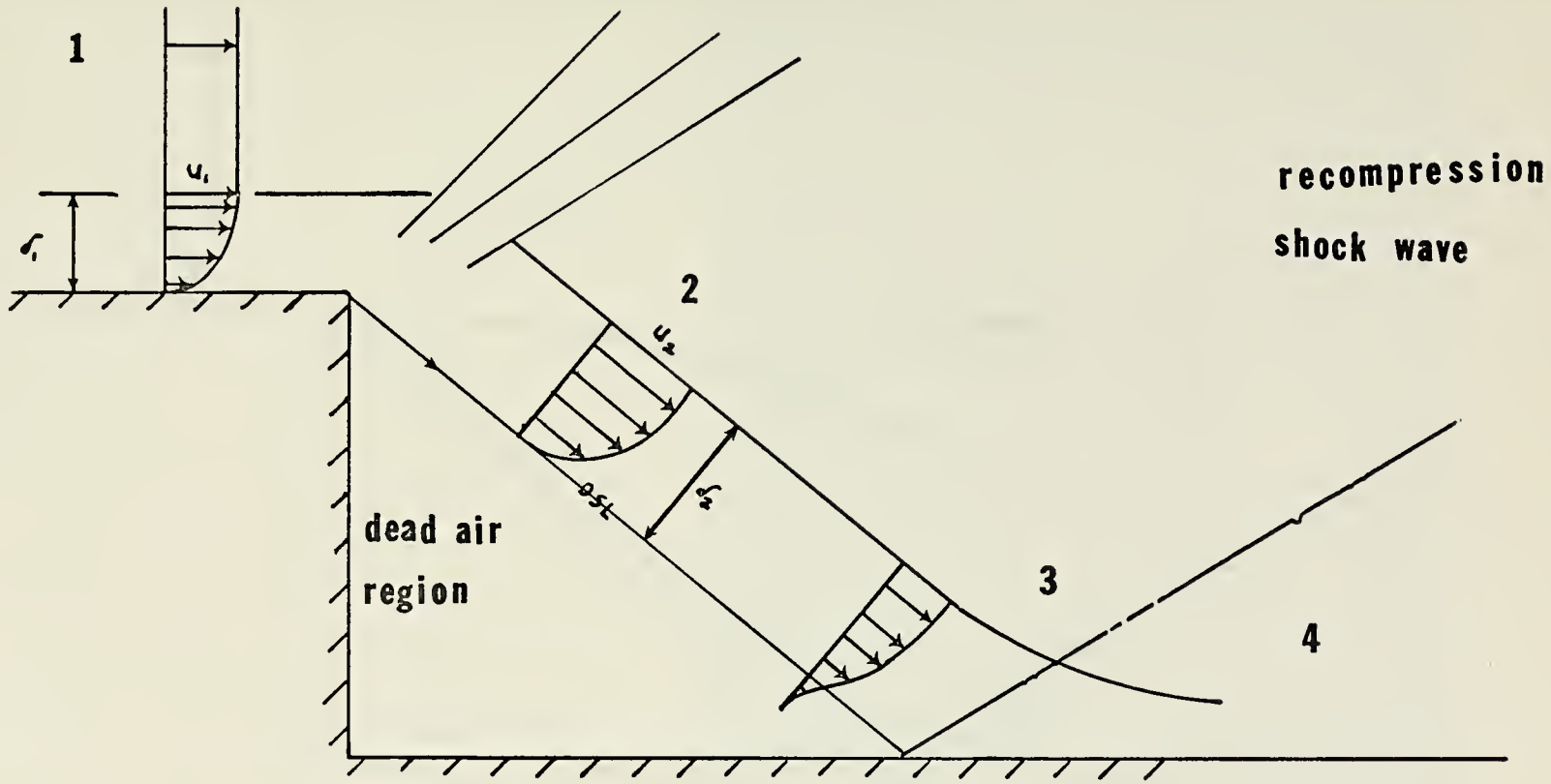


Figure 1. Chapman-Korst Flow Model for Supersonic Base Flow

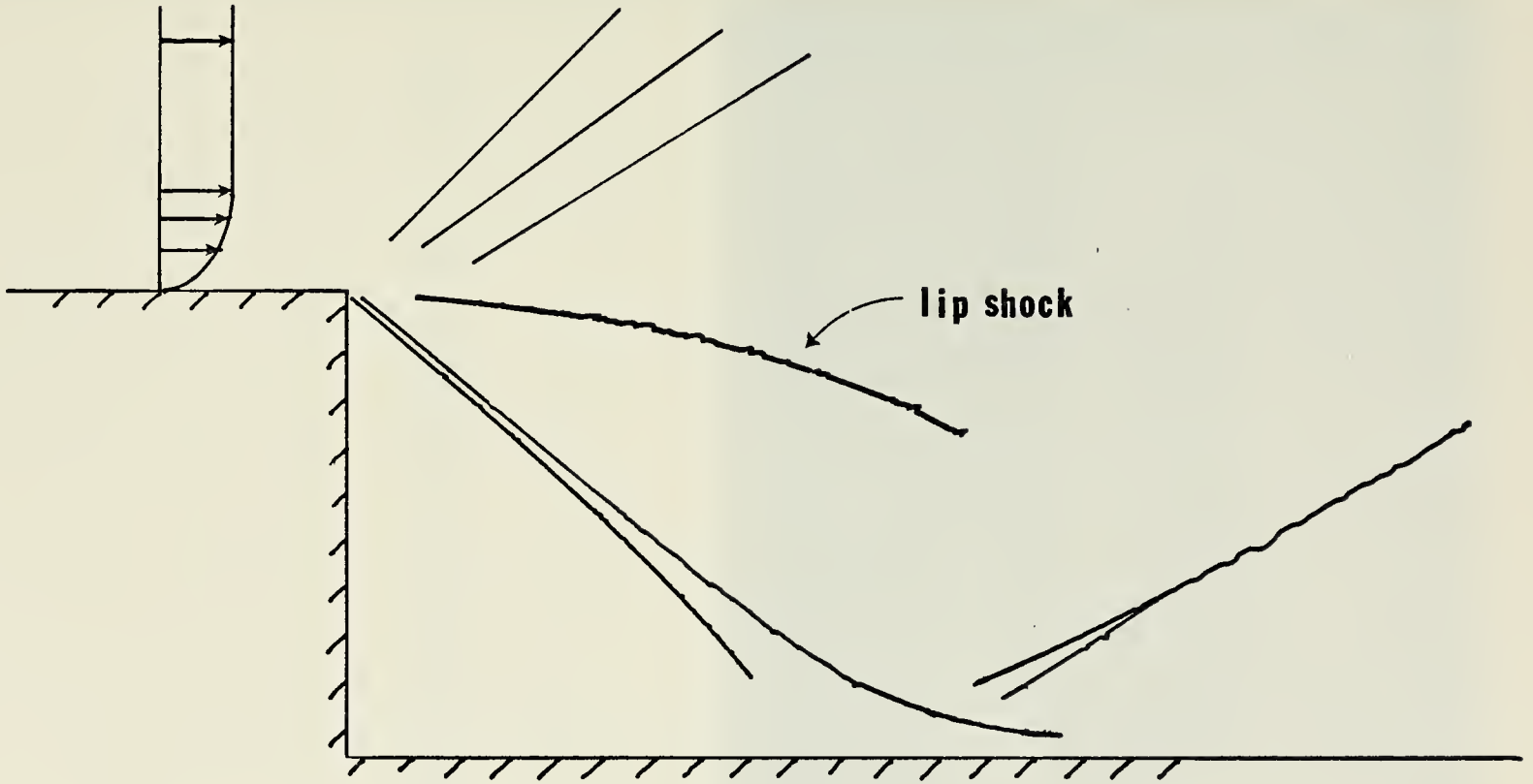


Figure 2. Lip Shock

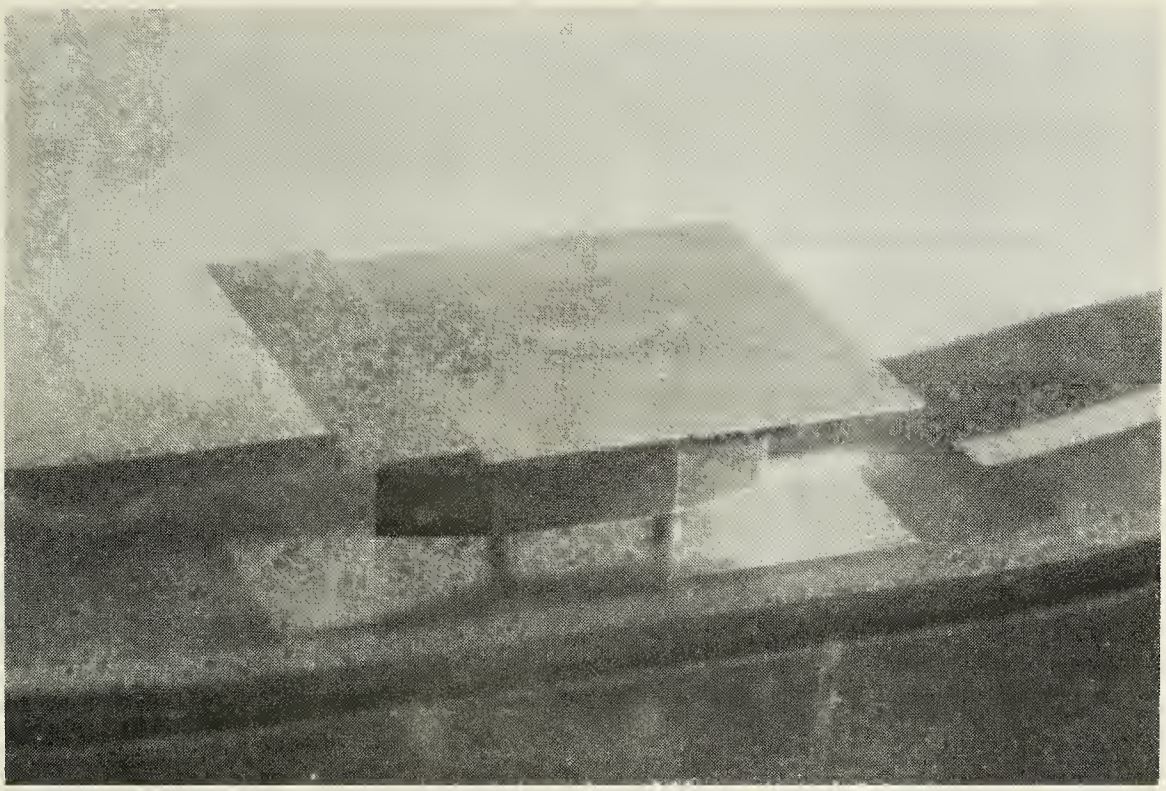


Figure 3. The Experimental Model of a Rear-Facing Step.

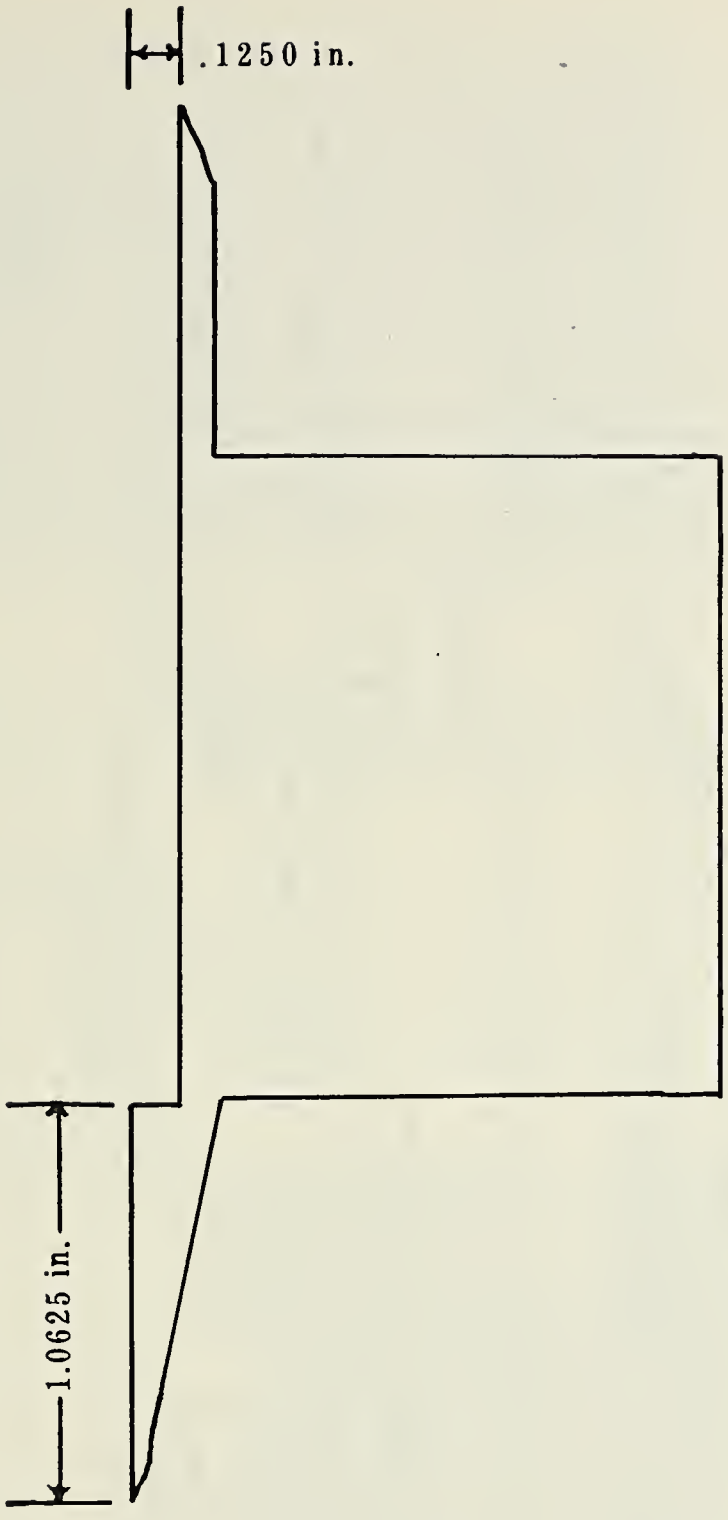


Figure 4. Experimental Model

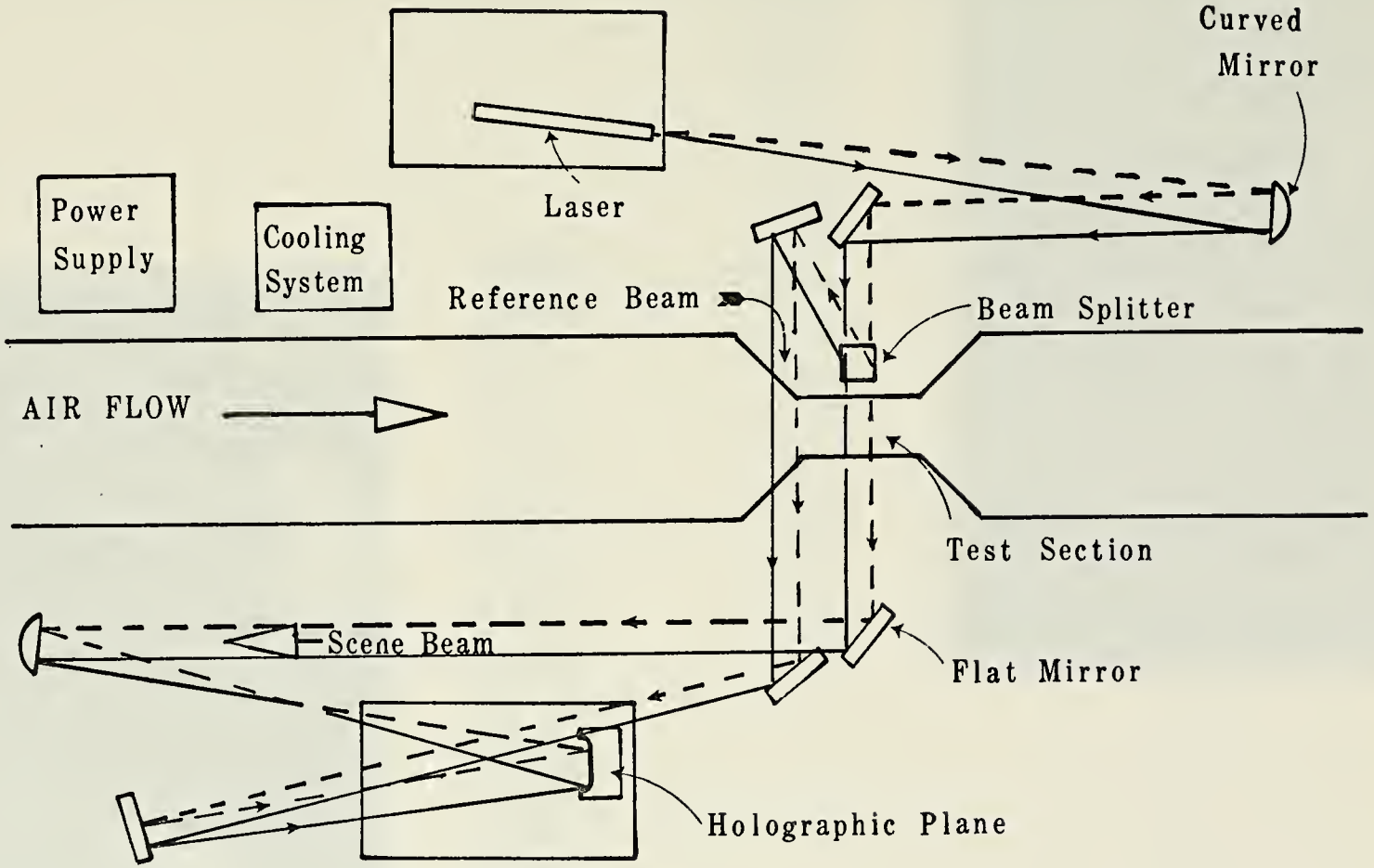


Figure 5. Experimental Setup

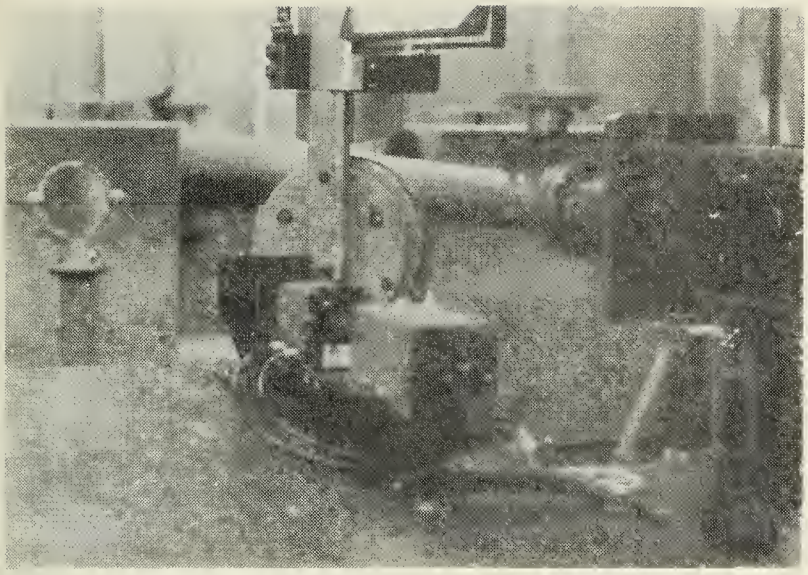
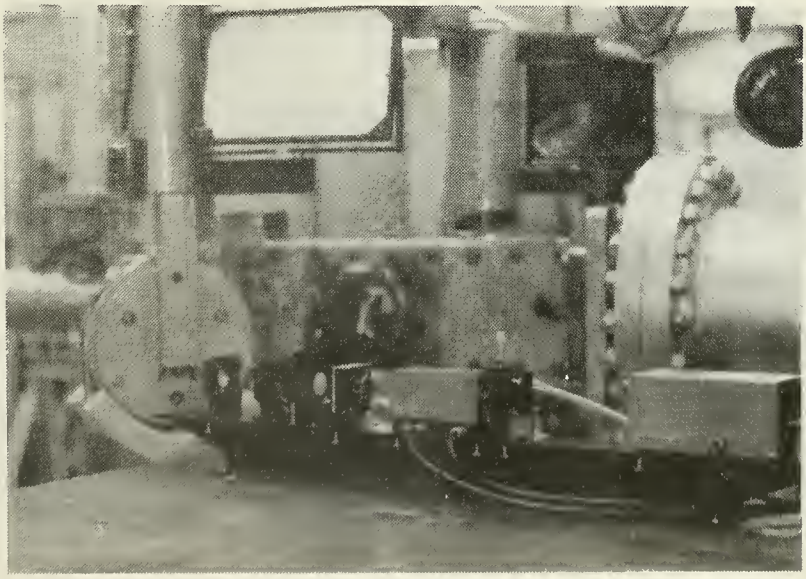


Figure 6. A view of the left side of the the experimental setup showing the laser and the optics system up to the beam splitter.

Figure 7. View of the left side of the test section, also showing the Q-switched laser, the beam splitter, and mirrors which direct the reference beam over the test section.



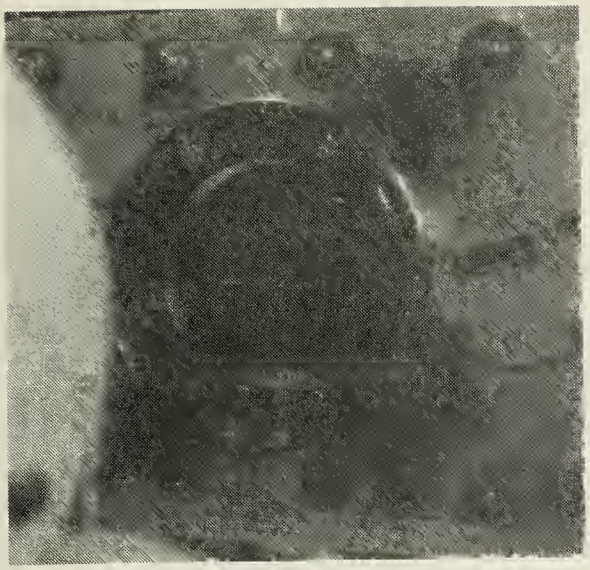
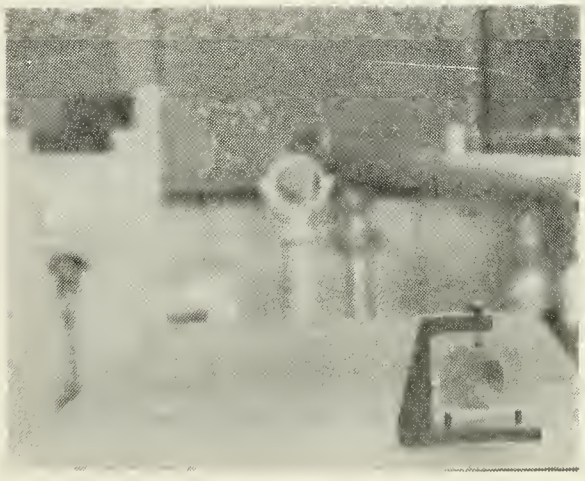


Figure 8.

A view of the right side of the test section.

Figure 9.

A view of the right side of the experimental setup showing the holographic plane and the final mirrors for the reference and the scene beams.



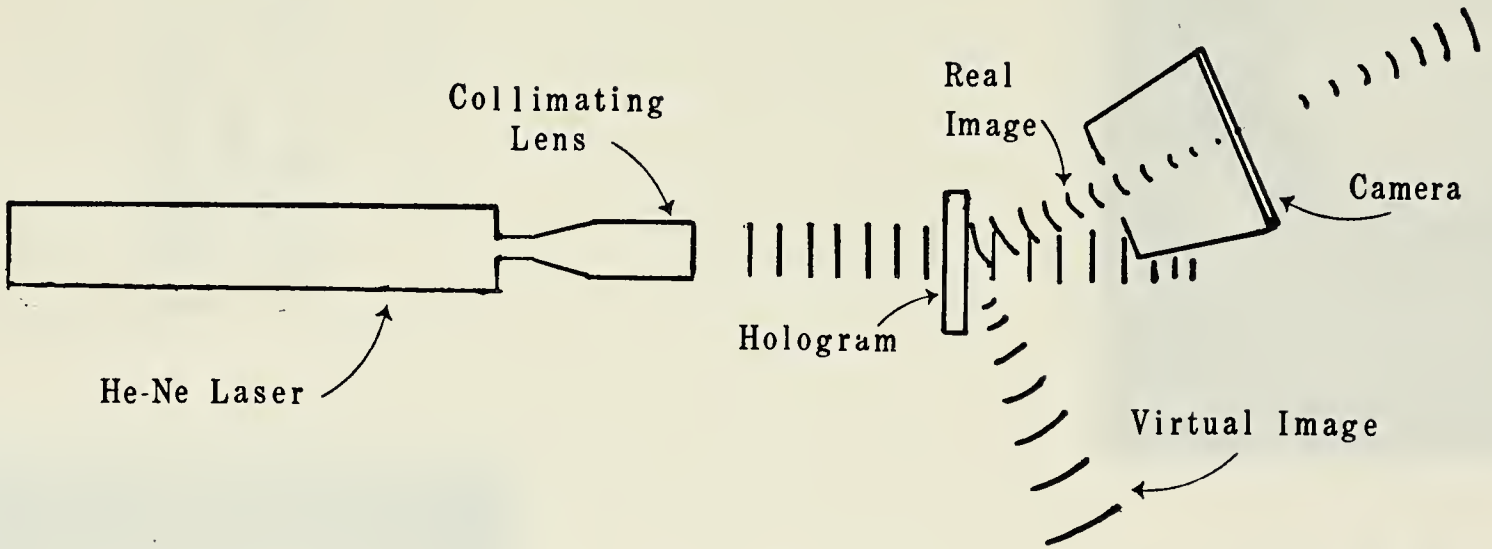


Figure 10. Reconstruction Setup

Figure 11.

Photograph of the projected real image of a single-exposed hologram.

Figure 12.

Photograph of the projected real image of a single-exposed hologram.

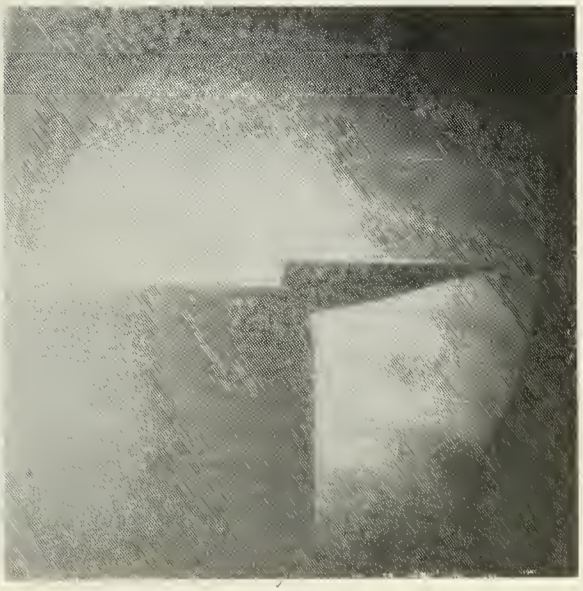
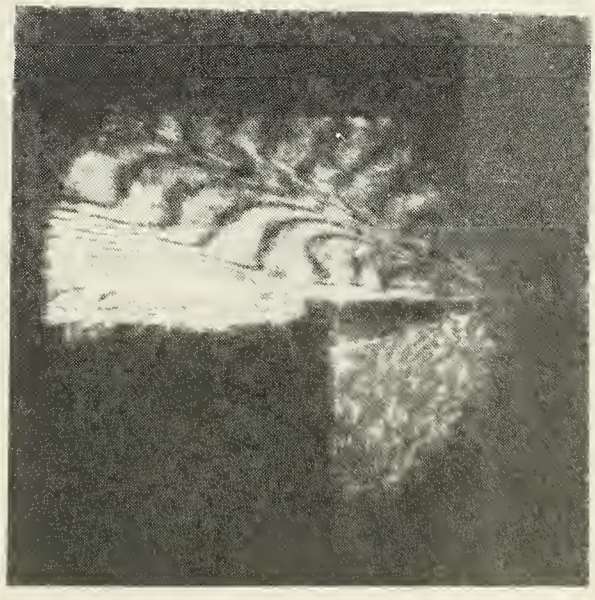


Figure 13.

Photograph of the projected real image of a double-exposed hologram of a rear-facing 1/8 in. step in Mach 2.8 flow.

Figure 14.

Photograph of the projected real image of a double-exposed hologram of a rear-facing 1/8 in. step in Mach 2.8 flow.



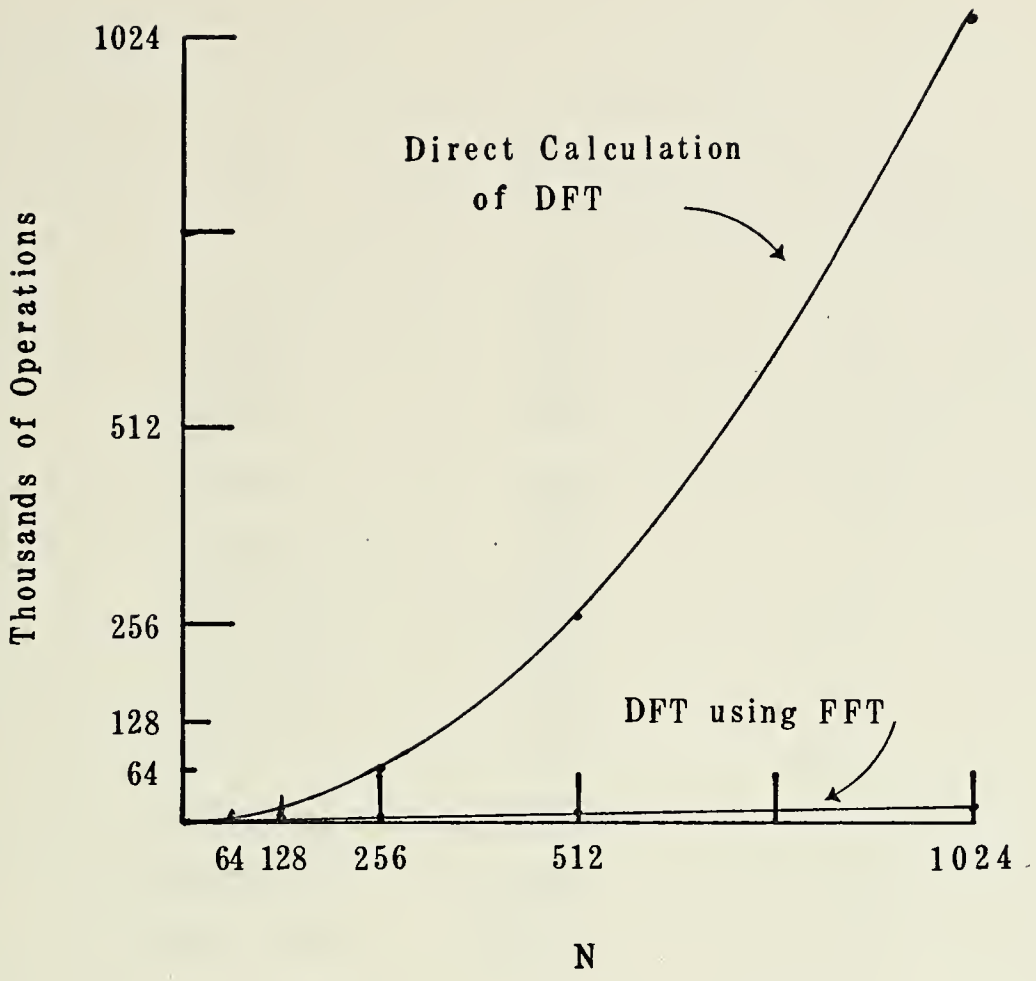


Figure 15. The Number of Operations Required to Compute the Discrete Fourier Transform by Direct Calculation Compared with Using the FFT Algorithm.

Q	N	γ
10	32	5
11-19	64	6
20-29	128	7
30-49	256	8
50-99	512	9
100-199	1024	10
200-299	2048	11
300-599	4096	12
600-999	8192	13
1000-1999	16384	14
2000-3999	32768	15

Optimum Value of N for FFT Convolution.

FIGURE 16.

APPENDIX A. DETAILED EXPERIMENTAL PROCEDURES

I. ALIGNMENT OF SYSTEM:

a) Turn on cooling system and warm up power supply.

b) Check internal alignment of laser with autocollimator.

c) Insert 2.5 mm orifice inside laser cavity in order to eliminate higher order transverse mode emission.

d) After ensuring all endangered personnel are wearing eye protectors, fire the Q-switch laser to burn a spot on underexposed (black) film.

e) Make a needle hole in the center of the burn spot and fire the laser again to make a burn spot on another fixed piece of underexposed film. Always check for personnel wearing eye protectors prior to firing the Q-switch laser.

f) Align a 15 mW CW laser beam through the 2.5 mm orifice and through the pin holes in the two pieces of film by using a flat mirror.

g) Elevate table so that the laser beam hits the first mirror in the optics system in its center. Ensure that the focal point of the diffusing lens is located at the focal point of the parabolic mirror in order to obtain a collimated beam entering the optics system.

h) Align optics system, ensuring test section and holographic plane are properly illuminated.

i) Adjust the location of the holographic plane to ensure that the subject image is in focus in the diverging part of the scene beam.

j) Adjust the final flat mirror in the reference beam in order to make sure path lengths of the two beams are within one-half inch of each other.

k) Remove CW laser and alignment mirror as well as remaining alignment pinholes.

l) Ready to fire.

II. EXPOSURE OF HOLOGRAPHIC PLATES:

a) Single Pulse (Exposure):

1) Insert plate in holder underneath the black drape.

2) Assistant charges the power supply and notifies experimentor when ready to fire.

3) Experimentor lifts the drape while simultaneously calling to fire the laser, and replaces the drape immediately after observing lamp flash of the Q-switch laser.

4) Remove the exposed holographic plate to a labeled film box while still under the drape to eliminate further exposure.

5) Hologram is ready for development.

b) Double-Pulse (Exposure):

1) Repeat steps II.a.1-3. Wear sound attenuators.

2) Open air supply and create desired flow in the test section. (Mach 2.8 flow in this experiment)

3) Repeat steps II.a.2-5. Assistant shuts off air supply after firing laser.

APPENDIX B. INSIDE INTEGRAL SUBSECTION OF VAN HOUTEN'S COMPUTER PROGRAM

See Appendix E right hand count labeled EVE02460-2840 for location of this subsection in Van Houten's program.

```
*****CALCULATE VALUE OF INSIDE INTEGRAL ****************************** ***03370
      DO 43 I=1,NT                                                     ***03390
      DO 43 J=1,NP                                                     ***03400
      M=NPL+J                                                          ***03410
      BD(1)=0.                                                         ***03420
      DO 39 K=1,MAX                                                    ***03430
      MKP=M+K                                                          ***03440
      MKM=M-K                                                          ***03450
      IF(MKP.GT.NPP) MKP=NPP                                           ***03460
      IF(MKM.LT.1) MKM=1                                               ***03470
      K1=K**2                                                          ***03480
   39 BD(K+1)=(BB(MKP,I)+BB(MKM,I)-2.*BB(M,I))/(FLOAT(K1)*DR)          ***03490
      VAL=0.                                                           ***03500
      K1=1                                                             ***03510
   40 K2=K1+6                                                          ***03520
      IF(K2.GT.MAX) GO TO 41                                           ***03530
*****COTES SIXTH ORDER QUADATURE FORMULA*****************               ***03540
      VAL=VAL+C1*(BD(K1)+BD(K2))+C2*(BD(K1+1)+BD(K2-1))+C3*(BD(K1+2)+BD(***03550
     1K2-2))+C4*BD(K1+3)                                               ***03560
      K1=K2                                                            ***03570
      GO TO 40                                                         ***03580
   41 SUM=0.                                                           ***03590
*****CONTRIBUTION OF LAST TERM*****************************************03600
      DO 42 K=K1,MAX                                                   ***03610
   42 SUM=SUM+BD(K)+BD(K+1)                                            ***03620
*****SUM COTES PLUS LAST TERMS PLUS INTEGRATION FROM 2 RMAX TO INFINITY***03630
   43 B(J,I)= VAL+SUM/2.-.666667*BB(M,I)/RMAX                          ***03640
      IF(KSYM.LT.99) GO TO 48                                          ***03650
```

APPENDIX C. FFT CONVOLUTION PROCEDURES

C.1 COMPUTATION PROCEDURE FOR FFT CONVOLUTION OF FINITE-LENGTH FUNCTIONS.

1. Let $x(t)$ and $h(t)$ be finite-length functions shifted from the origin by a and b, respectively.

2. Shift $x(t)$ and $h(t)$ to the origin and sample:

$$x(k) = x(kT + a) \qquad k = 0,1,\ldots,P-1$$

$$h(k) = h(kT + b) \qquad k = 0,1,\ldots,Q-1$$

3. Choose N to satisfy the relationships:

$$N \geq P + Q - 1$$

$$N = 2 \qquad \text{integer valued}$$

where P is the number of samples defining $x(t)$ and Q is the number of samples defining $h(t)$.

4. Augment with zeros the sampled functions of step (2).

$$x(k) = 0 \qquad k = P,P+1,\ldots,N-1$$

$$h(k) = 0 \qquad k = Q,Q+1,\ldots,N-1$$

5. Compute the discrete transform of $x(k)$ and $h(k)$.

$$X(n) = \sum_{k=0}^{N-1} x(k) \exp(-j2\pi nk/N)$$

$$H(n) = \sum_{k=0}^{N-1} h(k) \exp(-j2\pi nk/N)$$

6. Compute the product:

$$Y(n) = X(n)H(n)$$

7. Compute the inverse discrete transform using the forward transform:

$$y(k) = \sum_{n=0}^{N-1} \frac{1}{N} Y^*(n) \exp(-j2\pi nk/N)$$

C.2 COMPUTATION PROCEDURE FOR FFT CONVOLUTION: SELECT-SAVINGS METHOD:

1. Let Q be the number of samples representing h(t).

2. Choose N according to the table in Figure 16.

3. Form the sampled periodic function h(k).

$$h(k) = h(kT) \qquad k = 0,1,\ldots,Q-1$$

$$= 0 \qquad k = Q,Q+1,\ldots,N-1$$

4. Compute the discrete Fourier transform of h(k)

$$H(n) = \sum_{k=0}^{N-1} h(k)\ \exp(-j2\pi nk/N)$$

5. Form the sampled periodic function:

$$x_i(k) = x(kT) \qquad k = 0,1,\ldots,N-1$$

6. Compute the discrete Fourier transform of $x_i(k)$.

$$X_i(n) = \sum_{k=0}^{N-1} x_i(k)\ \exp(-j2\pi nk/N)$$

7. Compute the product:

$$Y_i(n) = X_i(n)H(n)$$

8. Compute the inverse discrete transform of $Y_i(n)$.

$$y_i(k) = \sum_{n=0}^{N-1} \frac{1}{N} Y_i(n) \# \exp(-j2\pi nk/N)$$

9. Delete samples $y_i(0), y_i(1), \ldots y_i(Q-2)$, and save the remaining samples.

10. Repeat steps 5 to 9 until all sections are computed.

11. Combine the sectioned results by the relationships:

$$y(k) \text{ undefined} \qquad k = 0,1,\ldots,Q-2$$

$$y(k) = y_i(k) \qquad k = Q-1,Q,\ldots,N-1$$

$$y(k+N) = y_2(k+Q-1) \qquad k = 0,1,\ldots,N-Q$$

$$y(k+2N) = y_3(k+Q-1) \qquad k = 0,1,\ldots,N-Q$$

.

.

C.3 COMPUTATION PROCEDURE FOR FFT CONVOLUTION: OVERLAP-ADD METHOD.

1. Let Q be the number of samples representing h(t).

2. Choose N according to the table in Figure 16.

3. Form the sampled periodic function h(k):

$$h(k) = h(kT) \qquad k = 0,1,\ldots, Q - 1$$

$$= 0 \qquad k = Q, Q + 1,\ldots, N - 1$$

4. Compute the discrete Fourier transform of h(k).

$$H(n) = \sum_{k=0}^{N-1} h(k) \exp(-j2\pi nk/N)$$

5. Form the sampled periodic function:

$$x_i(k) = x(kT) \qquad k = 0,1,\ldots, N - Q$$

$$= 0 \qquad k = N - Q + 1,\ldots, N - 1$$

6. Compute the discrete Fourier transform of $x_i(k)$.

$$X_i(n) = \sum_{k=0}^{N-1} x_i(k) \exp(-j2\pi nk/N)$$

7. Compute the product:

$$Y_i(n) = X_i(n)H(n)$$

8. Compute the inverse discrete transform of $Y_i(n)$.

$$y_i(k) = \sum_{n=0}^{N-1} \frac{1}{N} Y_i(n) \# \exp(-j2\pi nk/N)$$

9. Repeat steps 5-8 until all sections are computed.

10. Combine the sectioned results by the relationships:

$$y(k) = y_1(k) \qquad k = 0,1,\ldots,N - Q$$

$$y(k+N-Q+1) = y_1(k+N-Q+1) + y_2(k) \qquad k = 0,1,\ldots,N - Q$$

$$y(k+2(N-Q+1)) = y_2(k+N-Q+1) + y_3(k) \qquad k = 0,1,\ldots,N - Q$$

.

.

.

```
C     ..................................................................CFFT0010
C                                                                      CFFT0020
C     SUBROUTINE  CFFT2      (CATEGORY E2)                             CFFT0030
C                                                                      CFFT0040
C     PURPOSE                                                          CFFT0050
C                                                                      CFFT0060
C        TO COMPUTE THE COMPLEX FOURIER TRANSFORM OR INVERSE IN PLACE  CFFT0070
C     USING THE MIXED-RADIX FAST FOURIER TRANSFORM ALGORITHM. WITHIN   CFFT0080
C     CERTAIN LIMITATIONS, DISCUSSED BELOW, AN ARBITRARY NUMBER OF DATA CFFT0090
C     POINTS MAY BE PROCESSED. SEE ROUTINE RFFT2 TO PROCESS REAL DATA  CFFT0100
C     IF THE NUMBER OF DATA POINTS IS EVEN.                            CFFT0110
C                                                                      CFFT0120
C     USAGE                                                            CFFT0130
C                                                                      CFFT0140
C         CALL CFFT2 (A, B, NTOT, N, NSPAN, ISN)                       CFFT0150
C                                                                      CFFT0160
C     DESCRIPTION OF PARAMETERS                                        CFFT0170
C                                                                      CFFT0180
C         A - REAL*4 ARRAY HOLDING THE CONSECUTIVELY STORED REAL COM-  CFFT0190
C             PONENTS OF DATA (WHEN IABS(ISN) = 1) OR A COMPLEX*8      CFFT0200
C             ARRAY HOLDING DATA (WHEN IABS(ISN) = 2). AT EXIT FROM    CFFT0210
C             CFFT2, A WILL HOLD THE REAL COMPONENTS OF THE RESULTING  CFFT0220
C             REAL*4 FOURIER COEFFICIENTS WHEN IABS(ISN)=1, OR THE     CFFT0230
C             COMPLEX*8 FOURIER COEFFICIENTS, WHEN IABS(ISN) = 2.      CFFT0240
C             SEE DESCRIPTION OF 'ISN' BELOW.                          CFFT0250
C                                                                      CFFT0260
C         B - WHEN IABS(ISN)=1, B IS A REAL*4 ARRAY HOLDING THE CONSE- CFFT0270
C             CUTIVELY STORED IMAGINARY COMPONENTS OF DATA. AT EXIT    CFFT0280
C             FROM CFFT2 IT WILL CONTAIN THE IMAGINARY COMPONENTS OF   CFFT0290
C             THE FOURIER COEFFICIENTS.  WHEN USING COMPLEX*8 DATA     CFFT0300
C             (WHEN IABS(ISN) =2) ALWAYS USE 'A(2)' INSTEAD OF 'B' IN  CFFT0310
C             THE CALL STATEMENT, SINCE 'A' CONTAINS BOTH THE REAL AND CFFT0320
C             IMAGINARY COMPONENTS OF DATA. SEE 'ISN' BELOW.           CFFT0330
C                                                                      CFFT0340
C         NTOT - TOTAL NUMBER OF COMPLEX DATA VALUES INVOLVED. FOR A   CFFT0350
C             SINGLE-VARIABLE ANALYSIS 'NTOT'='N'='NSPAN'=(NUMBER      CFFT0360
C             OF COMPLEX DATA VALUES).  FOR A TRI-VARIATE ANALYSIS     CFFT0370
C             OF 'A(N1,N2,N3)' AND 'B(N1,N2,N3)' WHERE IABS(ISN) =1,   CFFT0380
C             OR COMPLEX 'A(N1,N2,N3)' WHERE IABS(ISN) = 2, 'NTOT'     CFFT0390
C             WOULD BE THE PRODUCT, N1*N2*N3. (INTEGER*4)              CFFT0400
C             SEE "EXAMPLES" BELOW.                                    CFFT0410
C                                                                      CFFT0420
```

```
C          N - DIMENSION OF CURRENT VARIABLE. IDENTICAL TO NTOT AND       CFFT0430
C              NSPAN AND THE NUMBER OF COMPLEX DATA VALUES                CFFT0440
C              IN SINGLE VARIABLE CASE. SEE "EXAMPLES" BELOW FOR CASES    CFFT0450
C              OF HIGHER DIMENSIONALITY. (INTEGER*4)                      CFFT0460
C                                                                         CFFT0470
C          NSPAN - NSPAN/N IS THE SPACING OF CONSECUTIVE DATA VALUES      CFFT0480
C              WHEN INDEXING CURRENT VARIABLE. IT IS IDENTICAL TO NTOT    CFFT0490
C              AND N AND NO. OF COMPLEX DATA POINTS IN THE SINGLE-VARI-   CFFT0500
C              ABLE CASE. SEE "EXAMPLES" BELOW FOR MULTI-VARIABLE CASES.  CFFT0510
C              (INTEGER*4)                                                CFFT0520
C                                                                         CFFT0530
C          ISN - AN INTEGER*4 CONTROL VARIABLE BY WHICH USER SPECIFIES    CFFT0540
C              BOTH THE STORAGE MODE OF HIS DATA (AND RESULTING FOURIER   CFFT0550
C              COEFFICIENTS) AND WHETHER HE WANTS THE DISCRETE FOURIER    CFFT0560
C              TRANSFORM (DFT) OR THE INVERSE DISCRETE FOURIER TRANSFORM  CFFT0570
C              (IDFT).                                                    CFFT0580
C                                                                         CFFT0590
C              IF ISN IS NEGATIVE THE DFT IS OBTAINED; IF ISN IS POSI-    CFFT0600
C              TIVE THE IDFT IS OBTAINED.  IF ISN IS +1 OR -1, THE REAL   CFFT0610
C              COMPONENTS OF THE DATA SHOULD BE IN THE REAL*4 ARRAY 'A'   CFFT0620
C              AND THE IMAGINARY COMPONENTS IN THE REAL*4 ARRAY 'B', BOTH CFFT0630
C              OF WHICH SHOULD BE DIMENSIONED AT LEAST 'NTOT'. IF ISN IS  CFFT0640
C              +2 OR -2, THE COMPLEX DATA SHOULD BE STORED IN THE COM-    CFFT0650
C              PLEX*8 ARRAY 'A' (AT LEAST NTOT IN LENGTH) AND THE SECOND  CFFT0660
C              ARGUMENT IN THE CALLING LIST SHOULD BE 'A(2)'.             CFFT0670
C                                                                         CFFT0680
C      REMARKS                                                            CFFT0690
C                                                                         CFFT0700
C          DEFINITIONS:  THE DISCRETE FOURIER TRANSFORM (DFT) OF          CFFT0710
C          X(J) FOR J=1(1)N, IS 1/N TIMES THE SERIES GIVEN BY:            CFFT0720
C                                                                         CFFT0730
C          Y(K) =  THE SUMMATION FOR J = 1, N OF THE TERMS,               CFFT0740
C                  X(J) * EXP (-2.*PI * I * J * K / N),                   CFFT0750
C                  FOR K = 1(1)N.                                         CFFT0760
C                                                                         CFFT0770
C                  (HERE "I" STANDS FOR SQRT(-1.) AND PI= 3.141593.)      CFFT0780
C                                                                         CFFT0790
C          THE INVERSE DISCRETE FOURIER TRANSFORM (IDFT) OF Y(J),J=1,N    CFFT0800
C          IS THE SUMMATION FOR K = 1(1)N, OF THE TERMS,                  CFFT0810
C                                                                         CFFT0820
C                Y(K) * EXP(2.*PI* I * J * K/N).                          CFFT0830
C                                                                         CFFT0840
C          IMPORTANT NOTE:  CFFT2 RETURNS THE UNNORMALIZED  DFT WHEN      CFFT0850
C          ISN .LT.0.  THE RESULT VECTOR MUST BE DIVIDED BY N (NUMBER     CFFT0860
C          OF COMPLEX DATA POINTS) TO PRODUCE THE NORMALIZED  DFT.        CFFT0870
C                                                                         CFFT0880
C          NOTES ON NUMBER OF DATA POINTS:                                CFFT0890
C                                                                         CFFT0900
```

```
C          THE MOST DISTINCTIVE FEATURE OF CFFT2 IS THAT IT ACCOMO-   CFFT0910
C     DATES AN ALMOST ARBITRARY NUMBER OF DATA POINTS, EVEN THOUGH   CFFT0920
C     IT EMPLOYS THE COOLEY-TUKEY FAST FOURIER TRANSFORM ALGOR-      CFFT0930
C     ITHM.  HOWEVER FOR CERTAIN LARGE PROBLEMS, MINOR ALTERATIONS   CFFT0940
C     TO THE SOURCE CODE MAY BE REQUIRED. AS IT STANDS CFFT2 WILL    CFFT0950
C     ACCOMODATE ANY N .LE. 1008. TO ANALYZE THE SITUATION FOR A     CFFT0960
C     LARGER PROBLEM THE USER SHOULD FACTOR HIS N INTO ITS PRIME     CFFT0970
C     FACTORS. THEN THE CONSTANT MAXF IN CFFT2 MUST BE CHANGED TO    CFFT0980
C     BE .GE. THE GREATEST PRIME FACTOR ON N (ONLY IF THIS IS        CFFT0990
C     .GT. 997).                                                     CFFT1000
C                                                                    CFFT1010
C          ALSO THE ARRAYS 'AT', 'CK', 'BT', AND 'SK' MUST BE        CFFT1020
C     RE-DIMENSIONED TO BE AT LEAST MAXF IN LENGTH.                  CFFT1030
C     THEN THE USER SHOULD COMPUTE K, WHICH IS DEFINED AS            CFFT1040
C     THE PRODUCT OF THE REMAINING FACTORS OF N AFTER ALL SQUARED    CFFT1050
C     FACTORS ARE STRUCK OUT. IF  K IS .GT. 1001, THE CONSTANT       CFFT1060
C     MAXP MUST BE ALTERED TO BE AT LEAST K-1, AND THE ARRAY NP      CFFT1070
C     MUST BE RE-DIMENSIONED TO MAXP. FINALLY IF THE NUMBER OF       CFFT1080
C     FACTORS (TO THE FIRST POWER) OF N IS .GT. 15, THE ARRAY NFAC   CFFT1090
C     MUST BE INCREASED TO THAT NUMBER. IF AVAILABLE STORAGE IS      CFFT1100
C     INSUFFICIENT, CFFT2 TERMINATES VIA A STOP 99 STATEMENT.        CFFT1110
C                                                                    CFFT1120
C     NOTE: FOR LARGE PROBLEMS REQUIRING ALTERATIONS DESCRIBED       CFFT1130
C     ABOVE, THE USER MAY HAVE TO SUBSTITUTE THE DOUBLE-PRECISION    CFFT1140
C     VERSION OF CFFT2, (DCFFT2).  SEE EXAMPLE BELOW FOR ACCURACY    CFFT1150
C     TEST.                                                          CFFT1160
C                                                                    CFFT1170
C EXAMPLES                                                           CFFT1180
C                                                                    CFFT1190
C     TO OBTAIN DFT OF SINGLE-VARIABLE  DATA STORED IN A             CFFT1200
C     COMPLEX*8  ARRAY, 'A':                                         CFFT1210
C           CALL CFFT2 (A, A(2), N, N, N, -2)                        CFFT1220
C                                                                    CFFT1230
C     TO OBTAIN THE IDFT OF SINGLE VARIABLE DATA STORED AT 'A' (REAL CFFT1240
C     COMPONENTS) AND  'B' (IMAGINARY COMPONENTS):                   CFFT1250
C           CALL CFFT2 (A, B, N, N, N, 1)                            CFFT1260
C                                                                    CFFT1270
C     TO OBTAIN THE DFT OF TRI-VARIATE DATA AT A(N1,N2,N3) (REAL     CFFT1280
C     COMPONENTS) AND B(N1,N2,N3) (IMAGINARY COMPONENTS):            CFFT1290
C                                                                    CFFT1300
C           CALL CFFT2 (A, B, N1*N2*N3, N1, N1, -1)                  CFFT1310
C           CALL CFFT2 (A, B, N1*N2*N3,  N2, N1*N2,-1)               CFFT1320
C           CALL CFFT2 (A, B, N1*N2*N3, N3, N1*N2*N3,-1)             CFFT1330
C                                                                    CFFT1340
C           NOTE: THREE CALLS OF CFFT2 ARE REQUIRED.                 CFFT1350
C                                                                    CFFT1360
C     TO CHECK ACCURACY ON SAMPLE SINGLE-VARIABLE COMPLEX DATA:      CFFT1370
C                                                                    CFFT1380
```

```
C               CALL CFFT2 (A, A(2), N, N, N, -2)                       CFFT1390
CCC             CALL CFFT2 (A, A(2), N, N, N,  2)                       CFFT1400
CCC             FN=  N                                                  CFFT1410
CCC             DO  33  I = 1, N                                        CFFT1420
CCC           33 A(I) =  A(I)/ FN                                       CFFT1430
CCC                                                                     CFFT1440
CCC        N.B. FINAL DATA IN 'A' WOULD BE IDENTICAL TO ORIGINAL DATA,  CFFT1450
CCC             IF NO ACCURACY WERE LOST. AN EXPERIMENTAL CALCULATION   CFFT1460
CCC             OF THIS SORT MAY REVEAL THE NECESSITY OF PERFORMING     CFFT1470
CCC             YOUR CALCULATIONS IN DOUBLE PRECISION. (SEE DCFFT2.)    CFFT1480
CCC                                                                     CFFT1490
CCC     SIZE: 53K BYTES, INCLUDING SYSTEM ROUTINES AND EXCLUDING DATA.  CFFT1500
CCC                                                                     CFFT1510
CCC     SUBROUTINES/FUNCTIONS REQUIRED                                  CFFT1520
CCC                                                                     CFFT1530
CCC         ONLY BUILT-IN FORTRAN FUNCTIONS ARE REQUIRED.               CFFT1540
CCC                                                                     CFFT1550
CCC     METHOD/REFERENCES                                               CFFT1560
CCC                                                                     CFFT1570
CCC         1.) SINGLETON, R.C., "AN ALGORITM FOR COMPUTING THE MIXED   CFFT1580
CCC             RADIX FAST FOURIER TRANSFORM," IEEE TRANSACTIONS ON     CFFT1590
CCC             AUDIO AND ELECTROACOUSTICS, 17(2), JUNE 1969, PP. 93-103. CFFT1600
CCC                                                                     CFFT1610
CCC         2.) RITEA, H.B., " A COMPARISON OF FFT ALGORITHMS, " SYS DEV CFFT1620
CCC             CORP,  5 JAN 1972, PP. 9-20. (AD 735-920) (TM4857/100). CFFT1630
CCC                                                                     CFFT1640
CCC     PROGRAMMER:  R. C. SINGLETON, S.R.I., OCT. 1968                 CFFT1650
CCC                                                                     CFFT1660
CCC     IMPLEMENTER AT NPS:  R. R. HILLEARY, MARCH 1973                 CFFT1670
CCC                                                                     CFFT1680
CCC ....................................................................CFFT1690
      SUBROUTINE  CFFT2(A,B,NTOT,N,NSPAN,ISN)                           CFFT1700
      DIMENSION A(N), B(N)                                              CFFT1710
      DIMENSION NFAC(15), NP(1000)                                      CFFT1720
      DIMENSION AT(1000), CK(1000), BT(1000), SK(1000)                  CFFT1730
      EQUIVALENCE (I,II)                                                CFFT1740
C                                                                       CFFT1750
CCC     THE FOLLOWING CONSTANT SHOULD MATCH ARRAYS' DIMENSION           CFFT1760
C                                                                       CFFT1770
      MAXF = 997                                                        CFFT1780
C       THE FOLLOWING CONSTANT MAY ALSO NEED ADJUSTMENT.  SEE WRITELP.  CFFT1790
      MAXP = 1000                                                       CFFT1800
C                                                                       CFFT1810
      IF (N .LT. 2) RETURN                                              CFFT1820
      INC = ISN                                                         CFFT1830
      RAD = 6.2831853ID0                                                CFFT1840
      S72 = RAD/5.                                                      CFFT1850
      C72 =  COS(S72)                                                   CFFT1860
```

```
      S72 =  SIN(S72)                                                   CFFT1870
C                                                                       CFFT1880
      S120 = .8660254D0                                                 CFFT1890
      IF (ISN  .GE. 0) GO TO 10                                         CFFT1900
      S72 = -S72                                                        CFFT1910
      S120 = -S120                                                      CFFT1920
      RAD = -RAD                                                        CFFT1930
      INC = -INC                                                        CFFT1940
C                                                                       CFFT1950
   10 NT = INC * NTOT                                                   CFFT1960
      KS = INC * NSPAN                                                  CFFT1970
      KSPAN = KS                                                        CFFT1980
      NN = NT - INC                                                     CFFT1990
      JC = KS/N                                                         CFFT2000
      RADF = RAD * FLOAT(JC) * .5                                       CFFT2010
      I= 0                                                              CFFT2020
      JF = 0                                                            CFFT2030
C                                                                       CFFT2040
C        DETERMINE FACTORS OF N                                         CFFT2050
C                                                                       CFFT2060
      M = 0                                                             CFFT2070
      K = N                                                             CFFT2080
      GO TO 20                                                          CFFT2090
C                                                                       CFFT2100
   15 M = M + 1                                                         CFFT2110
      NFAC(M) = 4                                                       CFFT2120
      K = K/16                                                          CFFT2130
C                                                                       CFFT2140
   20 IF (K - (K/16) * 16 .EQ. 0) GO TO 15                              CFFT2150
      J = 3                                                             CFFT2160
      JJ = 9                                                            CFFT2170
      GO TO 30                                                          CFFT2180
C                                                                       CFFT2190
   25 M = M + 1                                                         CFFT2200
      NFAC(M) = J                                                       CFFT2210
      K = K/JJ                                                          CFFT2220
C                                                                       CFFT2230
   30 IF (MOD(K,JJ) .EQ. 0) GO TO 25                                    CFFT2240
      J = J + 2                                                         CFFT2250
      JJ =  J**2                                                        CFFT2260
      IF (JJ .LE. K) GO TO 30                                           CFFT2270
      IF (K .GT. 4) GO TO 40                                            CFFT2280
      KT = M                                                            CFFT2290
      NFAC(M+1) = K                                                     CFFT2300
      IF (K .NE. 1)  M=M+1                                              CFFT2310
      GO TO 80                                                          CFFT2320
C                                                                       CFFT2330
   40 IF (K - (K/4)*4 .NE. 0) GO TO 50                                  CFFT2340
```

```
      M = M + 1                                                          CFFT2350
      NFAC(M) = 2                                                        CFFT2360
      K = K/4                                                            CFFT2370
C                                                                        CFFT2380
   50 KT = M                                                             CFFT2390
      J = 2                                                              CFFT2400
C                                                                        CFFT2410
   60 IF (MOD(K,J) .NE. 0) GO TO 70                                      CFFT2420
      M = M+1                                                            CFFT2430
      NFAC(M) = J                                                        CFFT2440
      K = K/J                                                            CFFT2450
C                                                                        CFFT2460
   70 J = ((J+1)/2)* 2 + 1                                               CFFT2470
      IF (J .LE. K) GO TO 60                                             CFFT2480
C                                                                        CFFT2490
   80 IF (KT .EQ. 0)  GO TO 100                                          CFFT2500
      J = KT                                                             CFFT2510
C                                                                        CFFT2520
   90 M = M+1                                                            CFFT2530
      NFAC(M) = NFAC(J)                                                  CFFT2540
      J = J-1                                                            CFFT2550
      IF (J .NE. 0) GO TO 90                                             CFFT2560
C                                                                        CFFT2570
C          COMPUTE FOURIER TRANSFORM                                     CFFT2580
C                                                                        CFFT2590
  100 SD = RADF/ FLOAT(KSPAN)                                            CFFT2600
      CD = 2.0 *  SIN(SD)**2                                             CFFT2610
      SD =  SIN(SD+SD)                                                   CFFT2620
      KK = 1                                                             CFFT2630
      I = I + 1                                                          CFFT2640
      IF (NFAC(I) .NE. 2) GO TO 400                                      CFFT2650
C                                                                        CFFT2660
C          TRANSFORM FOR FACTOR OF 2 (INCLUDING ROTATION FACTOR)         CFFT2670
C                                                                        CFFT2680
      KSPAN = KSPAN/2                                                    CFFT2690
      K1 = KSPAN + 2                                                     CFFT2700
C                                                                        CFFT2710
  210 K2 = KK + KSPAN                                                    CFFT2720
      AK = A(K2)                                                         CFFT2730
      BK = B(K2)                                                         CFFT2740
      A(K2) = A(KK) - AK                                                 CFFT2750
      B(K2) = B(KK) - BK                                                 CFFT2760
      A(KK) = A(KK) + AK                                                 CFFT2770
      B(KK) = B(KK) + BK                                                 CFFT2780
C                                                                        CFFT2790
      KK = K2 + KSPAN                                                    CFFT2800
      IF (KK .LE. NN) GO TO 210                                          CFFT2810
      KK = KK - NN                                                       CFFT2820
```

```
      IF (KK .LE. JC) GO TO 210                                          CFFT2830
      IF (KK .GT. KSPAN) GO TO 800                                       CFFT2840
C                                                                        CFFT2850
  220 C1 = 1.0  -CD                                                      CFFT2860
      S1 = SD                                                            CFFT2870
C                                                                        CFFT2880
  230 K2 = KK + KSPAN                                                    CFFT2890
      AK = A (KK) - A(K2)                                                CFFT2900
      BK = B(KK) - B(K2)                                                 CFFT2910
      A(KK) = A(KK) + A(K2)                                              CFFT2920
      B(KK) = B(KK) + B(K2)                                              CFFT2930
      A(K2) = C1 * AK - S1 * BK                                          CFFT2940
      B(K2) = S1 * AK + C1 * BK                                          CFFT2950
      KK = K2 + KSPAN                                                    CFFT2960
      IF (KK .LT. NT) GO TO 230                                          CFFT2970
      K2 = KK - NT                                                       CFFT2980
      C1 = -C1                                                           CFFT2990
      KK = K1 - K2                                                       CFFT3000
      IF (KK .GT. K2) GO TO 230                                          CFFT3010
      AK = C1 - (CD * C1 + SD * S1)                                      CFFT3020
      S1 = (SD * C1 - CD * S1) +S1                                       CFFT3030
C                                                                        CFFT3040
      C1 = 0.5/(AK**2 + S1**2) + 0.5                                     CFFT3050
      S1 = C1 * S1                                                       CFFT3060
      C1 = C1 * AK                                                       CFFT3070
      KK = KK + JC                                                       CFFT3080
      IF (KK .LT. K2) GO TO 230                                          CFFT3090
C                                                                        CFFT3100
      K1 = K1 + INC + INC                                                CFFT3110
      KK = (K1 - KSPAN)/2 + JC                                           CFFT3120
      IF (KK .LE. JC+JC) GO TO 220                                       CFFT3130
      GO TO 100                                                          CFFT3140
C                                                                        CFFT3150
C          TRANSFORM FOR FACTOR OF 3                                     CFFT3160
C                                                                        CFFT3170
  320 K1 = KK + KSPAN                                                    CFFT3180
      K2 = K1 + KSPAN                                                    CFFT3190
      AK = A(KK)                                                         CFFT3200
      BK = B(KK)                                                         CFFT3210
      AJ = A(K1) + A(K2)                                                 CFFT3220
      BJ = B(K1) + B(K2)                                                 CFFT3230
      A(KK) = AK + AJ                                                    CFFT3240
      B(KK) = BK + BJ                                                    CFFT3250
      AK = -0.5 * AJ + AK                                                CFFT3260
      BK = -0.5 * BJ + BK                                                CFFT3270
      AJ = (A(K1)- A(K2)) * S120                                         CFFT3280
      BJ = (B(K1) - B(K2)) * S120                                        CFFT3290
C                                                                        CFFT3300
```

```
      A(K1) = AK - BJ                                                   CFFT3310
      B(K1) = BK + AJ                                                   CFFT3320
      A(K2) = AK + BJ                                                   CFFT3330
      B(K2) = BK - AJ                                                   CFFT3340
      KK = K2 + KSPAN                                                   CFFT3350
      IF (KK .LT. NN) GO TO 320                                         CFFT3360
      KK = KK - NN                                                      CFFT3370
      IF (KK .LE. KSPAN) GO TO 320                                      CFFT3380
      GO TO 700                                                         CFFT3390
C                                                                       CFFT3400
C          TRANSFORM FOR FACTOR OF 4                                    CFFT3410
C                                                                       CFFT3420
  400 IF (NFAC(I) .NE. 4) GO TO 600                                     CFFT3430
      KSPNN = KSPAN                                                     CFFT3440
      KSPAN = KSPAN/4                                                   CFFT3450
C                                                                       CFFT3460
  410 C1 = 1.0                                                          CFFT3470
      S1 = 0.0                                                          CFFT3480
C                                                                       CFFT3490
  420 K1 = KK + KSPAN                                                   CFFT3500
      K2 = K1 + KSPAN                                                   CFFT3510
      K3 = K2 + KSPAN                                                   CFFT3520
      AKP = A(KK) + A(K2)                                               CFFT3530
      AKM = A(KK) - A(K2)                                               CFFT3540
      AJP = A(K1) + A(K3)                                               CFFT3550
      AJM = A(K1) - A(K3)                                               CFFT3560
      A(KK) = AKP + AJP                                                 CFFT3570
      AJP = AKP - AJP                                                   CFFT3580
C                                                                       CFFT3590
      BKP = B(KK) + B(K2)                                               CFFT3600
      BKM = B(KK) - B(K2)                                               CFFT3610
      BJP = B(K1) + B(K3)                                               CFFT3620
      BJM = B(K1) - B(K3)                                               CFFT3630
      B(KK) = BKP + BJP                                                 CFFT3640
      BJP = BKP - BJP                                                   CFFT3650
C                                                                       CFFT3660
      IF (ISN .LT. 0) GO TO 450                                         CFFT3670
      AKP = AKM - BJM                                                   CFFT3680
      AKM = AKM + BJM                                                   CFFT3690
C                                                                       CFFT3700
      BKP = BKM + AJM                                                   CFFT3710
      BKM = BKM - AJM                                                   CFFT3720
      IF (S1 .EQ. 0.0) GO TO 460                                        CFFT3730
C                                                                       CFFT3740
  430 A(K1) = AKP*C1 - BKP*S1                                           CFFT3750
      B(K1) = AKP*S1 + BKP*C1                                           CFFT3760
C                                                                       CFFT3770
      A(K2) = AJP*C2 - BJP*S2                                           CFFT3780
```

```
      B(K2) = AJP*S2 + BJP*C2                                          CFFT3790
C                                                                      CFFT3800
      A(K3) = AKM*C3 - BKM*S3                                          CFFT3810
      B(K3) = AKM*S3 + BKM*C3                                          CFFT3820
C                                                                      CFFT3830
      KK = K3 + KSPAN                                                  CFFT3840
      IF (KK .LE. NT) GO TO 420                                        CFFT3850
C                                                                      CFFT3860
  440 C2 = C1 - (CD*C1 + SD*S1)                                        CFFT3870
      S1 = (SD*C1 - CD*S1) + S1                                        CFFT3880
C                                                                      CFFT3890
      C1 = 0.5/(C2**2 + S1**2) + 0.5                                   CFFT3900
      S1 = C1 * S1                                                     CFFT3910
      C1 = C1 * C2                                                     CFFT3920
      C2 = C1**2 - S1**2                                               CFFT3930
      S2 = 2.0 * C1 * S1                                               CFFT3940
      C3 = C2 * C1 - S2 * S1                                           CFFT3950
      S3 = C2 * S1 + S2 * C1                                           CFFT3960
C                                                                      CFFT3970
      KK = KK - NT + JC                                                CFFT3980
      IF (KK .LE. KSPAN) GO TO 420                                     CFFT3990
C                                                                      CFFT4000
      KK = KK - KSPAN + INC                                            CFFT4010
      IF (KK .LE. JC) GO TO 410                                        CFFT4020
C                                                                      CFFT4030
      IF (KSPAN .EQ. JC) GO TO 800                                     CFFT4040
      GO TO 100                                                        CFFT4050
C                                                                      CFFT4060
  450 AKP = AKM + BJM                                                  CFFT4070
      AKM = AKM - BJM                                                  CFFT4080
C                                                                      CFFT4090
      BKP = BKM - AJM                                                  CFFT4100
      BKM = BKM + AJM                                                  CFFT4110
      IF (S1 .NE. 0.0) GO TO 430                                       CFFT4120
C                                                                      CFFT4130
  460 A(K1) = AKP                                                      CFFT4140
      B(K1) = BKP                                                      CFFT4150
      A(K2) = AJP                                                      CFFT4160
      B(K2) = BJP                                                      CFFT4170
      A(K3) = AKM                                                      CFFT4180
      B(K3) = BKM                                                      CFFT4190
C                                                                      CFFT4200
      KK = K3 + KSPAN                                                  CFFT4210
      IF (KK .LE. NT) GO TO 420                                        CFFT4220
      GO TO 440                                                        CFFT4230
C                                                                      CFFT4240
C         TRANSFORM FOR A FACTOR OF 5.                                 CFFT4250
C                                                                      CFFT4260
```

```
  510 C2 = C72**2  - S72**2                                          CFFT4270
      S2 = 2.0 * C72 * S72                                           CFFT4280
C                                                                    CFFT4290
  520 K1 = KK + KSPAN                                                CFFT4300
      K2 = K1 + KSPAN                                                CFFT4310
      K3 = K2 + KSPAN                                                CFFT4320
      K4 = K3 + KSPAN                                                CFFT4330
C                                                                    CFFT4340
      AKP = A(K1) + A(K4)                                            CFFT4350
      AKM = A(K1) - A(K4)                                            CFFT4360
      BKP = B(K1) + B(K4)                                            CFFT4370
      BKM = B(K1) - B(K4)                                            CFFT4380
C                                                                    CFFT4390
      AJP = A(K2) + A(K3)                                            CFFT4400
      AJM = A(K2) - A(K3)                                            CFFT4410
      BJP = B(K2) + B(K3)                                            CFFT4420
      BJM = B(K2) - B(K3)                                            CFFT4430
C                                                                    CFFT4440
      AA = A(KK)                                                     CFFT4450
      BB = B(KK)                                                     CFFT4460
      A(KK) = AA + AKP + AJP                                         CFFT4470
      B(KK) = BB + BKP + BJP                                         CFFT4480
C                                                                    CFFT4490
      AK = AKP * C72  + AJP * C2  + AA                               CFFT4500
      BK = BKP * C72  + BJP * C2  + BB                               CFFT4510
      AJ = AKM * S72  + AJM * S2                                     CFFT4520
      BJ = BKM * S72 + BJM * S2                                      CFFT4530
C                                                                    CFFT4540
      A(K1) = AK - BJ                                                CFFT4550
      A(K4) = AK + BJ                                                CFFT4560
      B(K1) = BK + AJ                                                CFFT4570
      B(K4) = BK - AJ                                                CFFT4580
C                                                                    CFFT4590
      AK = AKP * C2 + AJP * C72 + AA                                 CFFT4600
      BK = BKP * C2 + BJP * C72 + BB                                 CFFT4610
      AJ = AKM * S2 - AJM * S72                                      CFFT4620
      BJ = BKM * S2 - BJM * S72                                      CFFT4630
C                                                                    CFFT4640
      A(K2) = AK - BJ                                                CFFT4650
      A(K3) = AK + BJ                                                CFFT4660
      B(K2) = BK + AJ                                                CFFT4670
      B(K3) = BK - AJ                                                CFFT4680
      KK = K4 + KSPAN                                                CFFT4690
C                                                                    CFFT4700
      IF (KK .LT. NN) GO TO 520                                      CFFT4710
      KK = KK - NN                                                   CFFT4720
C                                                                    CFFT4730
      IF (KK .LE. KSPAN) GO TO 520                                   CFFT4740
```

```
      GO TO 700                                                          CFFT4750
C                                                                        CFFT4760
C          TRANSFORM FOR ODD FACTORS.                                    CFFT4770
C                                                                        CFFT4780
  600 K = NFAC(I)                                                        CFFT4790
      KSPNN = KSPAN                                                      CFFT4800
      KSPAN = KSPAN/K                                                    CFFT4810
      IF (K .EQ. 3) GO TO 320                                            CFFT4820
C                                                                        CFFT4830
      IF (K .EQ. 5) GO TO 510                                            CFFT4840
C                                                                        CFFT4850
      IF (K .EQ. JF) GO TO 640                                           CFFT4860
C                                                                        CFFT4870
      JF = K                                                             CFFT4880
      S1 = RAD/FLOAT(K)                                                  CFFT4890
      C1 = COS(S1)                                                       CFFT4900
      S1 = SIN(S1)                                                       CFFT4910
      IF (JF .GT. MAXF) GO TO 998                                        CFFT4920
C                                                                        CFFT4930
      CK(JF) = 1.0                                                       CFFT4940
      SK(JF) = 0.0                                                       CFFT4950
      J = 1                                                              CFFT4960
C                                                                        CFFT4970
  630 CK(J) = CK(K) * C1 + SK(K) * S1                                    CFFT4980
      SK(J) = CK(K) * S1 - SK(K) * C1                                    CFFT4990
C                                                                        CFFT5000
      K = K - 1                                                          CFFT5010
      CK(K) =  CK(J)                                                     CFFT5020
      SK(K) = -SK(J)                                                     CFFT5030
C                                                                        CFFT5040
      J = J + 1                                                          CFFT5050
      IF (J .LT. K) GO TO 630                                            CFFT5060
C                                                                        CFFT5070
  640 K1 = KK                                                            CFFT5080
      K2 = KK + KSPNN                                                    CFFT5090
C                                                                        CFFT5100
      AA = A(KK)                                                         CFFT5110
      BB = B(KK)                                                         CFFT5120
      AK = AA                                                            CFFT5130
      BK = BB                                                            CFFT5140
C                                                                        CFFT5150
      J = 1                                                              CFFT5160
      K1 = K1 + KSPAN                                                    CFFT5170
C                                                                        CFFT5180
  650 K2 = K2 - KSPAN                                                    CFFT5190
      J = J + 1                                                          CFFT5200
C                                                                        CFFT5210
      AT(J) = A(K1) + A(K2)                                              CFFT5220
```

```
      AK =  AT(J) + AK                                                  CFFT5230
C                                                                       CFFT5240
      BT(J) = B(K1) + B(K2)                                             CFFT5250
      BK = BT(J) + BK                                                   CFFT5260
      J = J+1                                                           CFFT5270
C                                                                       CFFT5280
      AT(J) = A(K1) - A(K2)                                             CFFT5290
      BT(J) = B(K1) - B(K2)                                             CFFT5300
C                                                                       CFFT5310
      K1 = K1 + KSPAN                                                   CFFT5320
      IF (K1 .LT. K2) GO TO 650                                         CFFT5330
C                                                                       CFFT5340
      A(KK) = AK                                                        CFFT5350
      B(KK) = BK                                                        CFFT5360
C                                                                       CFFT5370
      K1 = KK                                                           CFFT5380
      K2 = KK+ KSPNN                                                    CFFT5390
      J = 1                                                             CFFT5400
C                                                                       CFFT5410
  66C K1 = K1 + KSPAN                                                   CFFT5420
      K2 = K2 - KSPAN                                                   CFFT5430
      JJ = J                                                            CFFT5440
C                                                                       CFFT5450
      AK = AA                                                           CFFT5460
      BK = BB                                                           CFFT5470
      AJ = 0.0                                                          CFFT5480
      BJ = 0.0                                                          CFFT5490
      K = 1                                                             CFFT5500
C                                                                       CFFT5510
  670 K = K + 1                                                         CFFT5520
      AK = AT(K) * CK(JJ) + AK                                          CFFT5530
      BK = BT(K) * CK(JJ) + BK                                          CFFT5540
C                                                                       CFFT5550
      K = K + 1                                                         CFFT5560
      AJ = AT(K) * SK(JJ) + AJ                                          CFFT5570
      BJ = BT(K) * SK (JJ) + BJ                                         CFFT5580
C                                                                       CFFT5590
      JJ = JJ + J                                                       CFFT5600
      IF (JJ .GT. JF)  JJ = JJ - JF                                     CFFT5610
C                                                                       CFFT5620
      IF (K .LT. JF)  GO TO 670                                         CFFT5630
C                                                                       CFFT5640
      K = JF - J                                                        CFFT5650
      A(K1) = AK - BJ                                                   CFFT5660
      B(K1) = BK + AJ                                                   CFFT5670
C                                                                       CFFT5680
      A(K2) = AK + BJ                                                   CFFT5690
      B(K2) = BK - AJ                                                   CFFT5700
```

```
C                                                                       CFFT5710
      J = J + 1                                                         CFFT5720
      IF (J .LT. K) GO TO 660                                           CFFT5730
C                                                                       CFFT5740
      KK = KK + KSPNN                                                   CFFT5750
      IF (KK .LE.NN) GO TO 640                                          CFFT5760
C                                                                       CFFT5770
      KK = KK - NN                                                      CFFT5780
      IF (KK .LE. KSPAN) GO TO 640                                      CFFT5790
C                                                                       CFFT5800
C          MULTIPLY BY ROTATION FACTOR (EXCEPT FOR FACTORS OF 2 AND 4). CFFT5810
C                                                                       CFFT5820
  700 IF (I .EQ.M) GO TO 800                                            CFFT5830
      KK = JC + 1                                                       CFFT5840
C                                                                       CFFT5850
  710 C2 = 1.0 - CD                                                     CFFT5860
      S1 = SD                                                           CFFT5870
C                                                                       CFFT5880
  720 C1 = C2                                                           CFFT5890
      S2 = S1                                                           CFFT5900
      KK = KK + KSPAN                                                   CFFT5910
C                                                                       CFFT5920
  730 AK = A(KK)                                                        CFFT5930
      A(KK) = C2 * AK - S2 * B(KK)                                      CFFT5940
      B(KK) = S2 * AK + C2 * B(KK)                                      CFFT5950
C                                                                       CFFT5960
      KK = KK + KSPNN                                                   CFFT5970
      IF(KK .LE.NT) GO TO 730                                           CFFT5980
C                                                                       CFFT5990
      AK = S1 * S2                                                      CFFT6000
      S2 = S1 * C2 + C1 * S2                                            CFFT6010
      C2 = C1 * C2 - AK                                                 CFFT6020
      KK = KK - NT + KSPAN                                              CFFT6030
      IF (KK .LE. KSPNN) GO TO 730                                      CFFT6040
C                                                                       CFFT6050
      C2 = C1 - (CD * C1 + SD * S1)                                     CFFT6060
      S1 = S1 + (SD * C1 - CD * S1)                                     CFFT6070
C                                                                       CFFT6080
      C1 = 0.5/(C2**2 + S1**2) +0.5                                     CFFT6090
      S1 = C1 * S1                                                      CFFT6100
      C2 = C1 * C2                                                      CFFT6110
C                                                                       CFFT6120
      KK = KK - KSPNN + JC                                              CFFT6130
      IF (KK .LE. KSPAN) GO TO 720                                      CFFT6140
C                                                                       CFFT6150
      KK = KK - KSPAN + JC + INC                                        CFFT6160
      IF (KK .LE. JC + JC) GO TO 710                                    CFFT6170
      GO TO 100                                                         CFFT6180
```

```
C                                                                       CFFT6190
C          PERMUTE RESULTS TO NORMAL ORDER IN TWO STAGES:               CFFT6200
C             PERMUTATION FOR SQUARE FACTORS OF N                       CFFT6210
C                                                                       CFFT6220
  800 NP(1) = KS                                                        CFFT6230
      IF (KT .EQ. 0) GO TO 890                                          CFFT6240
C                                                                       CFFT6250
      K = KT + KT + 1                                                   CFFT6260
      IF (M .LT. K) K = K - 1                                           CFFT6270
C                                                                       CFFT6280
      J = 1                                                             CFFT6290
      NP(K+1) = JC                                                      CFFT6300
C                                                                       CFFT6310
  810 NP(J+1) = NP(J)/NFAC(J)                                           CFFT6320
      NP(K) = NP(K+1) * NFAC(J)                                         CFFT6330
C                                                                       CFFT6340
      J = J + 1                                                         CFFT6350
      K = K - 1                                                         CFFT6360
      IF (J .LT. K)  GO TO 810                                          CFFT6370
C                                                                       CFFT6380
      K3 = NP(K+1)                                                      CFFT6390
      KSPAN = NP(2)                                                     CFFT6400
      KK = JC + 1                                                       CFFT6410
      K2 = KSPAN + 1                                                    CFFT6420
C                                                                       CFFT6430
      J = 1                                                             CFFT6440
      IF (N .NE. NTOT) GO TO 850                                        CFFT6450
C                                                                       CFFT6460
C             PERMUTATION  FOR SINGE-VARIATE TRANSFORM                  CFFT6470
C                                                                       CFFT6480
  820 AK = A(KK)                                                        CFFT6490
      A(KK) = A(K2)                                                     CFFT6500
      A(K2) = AK                                                        CFFT6510
C                                                                       CFFT6520
      BK =  B(KK)                                                       CFFT6530
      B(KK) = B(K2)                                                     CFFT6540
      B(K2) = BK                                                        CFFT6550
C                                                                       CFFT6560
      KK = KK + INC                                                     CFFT6570
      K2 = KSPAN + K2                                                   CFFT6580
      IF (K2 .LT. KS) GO TO 820                                         CFFT6590
C                                                                       CFFT6600
  830 K2 = K2 - NP(J)                                                   CFFT6610
      J = J + 1                                                         CFFT6620
      K2 = NP(J+1)  + K2                                                CFFT6630
      IF (K2 .GT. NP(J)) GO TO 830                                      CFFT6640
      J = 1                                                             CFFT6650
C                                                                       CFFT6660
```

```
  840 IF (KK .LT. K2) GO TO 820                                          CFFT6670
C                                                                        CFFT6680
      KK = KK + INC                                                      CFFT6690
      K2 = KSPAN + K2                                                    CFFT6700
      IF (K2 .LT. KS)  GO TO 840                                         CFFT6710
C                                                                        CFFT6720
      IF (KK .LT.KS) GO TO 830                                           CFFT6730
C                                                                        CFFT6740
      JC = K3                                                            CFFT6750
      GO TO 890                                                          CFFT6760
C                                                                        CFFT6770
C             PERMUTATION FOR MULTIVARIATE TRANSFORM                     CFFT6780
C                                                                        CFFT6790
  850 K = KK + JC                                                        CFFT6800
C                                                                        CFFT6810
  860 AK = A(KK)                                                         CFFT6820
      A(KK) = A(K2)                                                      CFFT6830
      A(K2) = AK                                                         CFFT6840
C                                                                        CFFT6850
      BK = B(KK)                                                         CFFT6860
      B(KK) = B(K2)                                                      CFFT6870
      B(K2) = BK                                                         CFFT6880
C                                                                        CFFT6890
      KK = KK + INC                                                      CFFT6900
      K2 = K2 + INC                                                      CFFT6910
      IF  (KK .LT. K) GO TO 860                                          CFFT6920
C                                                                        CFFT6930
      KK = KK + KS - JC                                                  CFFT6940
      K2 = K2 + KS - JC                                                  CFFT6950
      IF (KK .LT. NT) GO TO 850                                          CFFT6960
C                                                                        CFFT6970
      K2 = K2 - NT + KSPAN                                               CFFT6980
      KK = KK - NT + JC                                                  CFFT6990
      IF (K2 .LT. KS) GO TO 850                                          CFFT7000
C                                                                        CFFT7010
  870 K2 = K2 - NP(J)                                                    CFFT7020
      J = J + 1                                                          CFFT7030
C                                                                        CFFT7040
      K2 = NP(J+1) + K2                                                  CFFT7050
      IF (K2 .GT. NP(J))  GO TO 870                                      CFFT7060
      J = 1                                                              CFFT7070
C                                                                        CFFT7080
  880 IF (KK .LT. K2) GO TO 850                                          CFFT7090
      KK = KK + JC                                                       CFFT7100
      K2 = KSPAN + K2                                                    CFFT7110
C                                                                        CFFT7120
      IF (K2 .LT.KS) GO TO 880                                           CFFT7130
C                                                                        CFFT7140
```

```
      IF (KK .LT.KS) GO TO 870                                          CFFT7150
      JC = K3                                                           CFFT7160
C                                                                       CFFT7170
  85C IF (2 * KT + 1  .GE. M) RETURN                                    CFFT7180
C                                                                       CFFT7190
      KSPNN = NP(KT+1)                                                  CFFT72CC
C                                                                       CFFT7210
C             PERMUTATION FOR SQUARE-FREE FACTORS CF N.                 CFFT7220
C                                                                       CFFT7230
      J = M - KT                                                        CFFT7240
      NFAC(J+1) = 1                                                     CFFT7250
C                                                                       CFFT7260
  9CC NFAC(J) = NFAC(J) * NFAC(J+1)                                     CFFT7270
      J = J - 1                                                         CFFT7280
      IF (J .NE. KT) GO TC 900                                          CFFT7290
C                                                                       CFFT7300
      KT = KT + 1                                                       CFFT7310
      NN = NFAC(KT) - 1                                                 CFFT7320
      IF (NN .GT. MAXP) GO TO 998                                       CFFT7330
C                                                                       CFFT7340
      JJ = 0                                                            CFFT7350
      J = 0                                                             CFFT7360
      GC TO 906                                                         CFFT7370
C                                                                       CFFT7380
  902 JJ = JJ - K2                                                      CFFT7390
      K2 = KK                                                           CFFT7400
      K = K + 1                                                         CFFT7410
      KK = NFAC(K)                                                      CFFT7420
C                                                                       CFFT7430
  904 JJ = KK + JJ                                                      CFFT7440
      IF (JJ .GE. K2) GO TO 902                                         CFFT7450
C                                                                       CFFT7460
      NP(J) = JJ                                                        CFFT7470
C                                                                       CFFT7480
  906 K2 = NFAC(KT)                                                     CFFT7490
      K = KT + 1                                                        CFFT7500
      KK = NFAC(K)                                                      CFFT7510
C                                                                       CFFT7520
      J = J + 1                                                         CFFT7530
      IF (J .LE. NN) GO TO 904                                          CFFT7540
C                                                                       CFFT7550
C         DETERMINE PERMUTATION CYCLES CF LENGTH .GT. 1                 CFFT7560
C                                                                       CFFT7570
      J = 0                                                             CFFT7580
      GO TO 914                                                         CFFT7590
C                                                                       CFFT7600
  910 K = KK                                                            CFFT7610
      KK = NP(K)                                                        CFFT7620
```

```
      NP(K) = -KK                                                       CFFT7630
      IF (KK .NE. J)  GO TO 910                                         CFFT7640
      K 3 = KK                                                          CFFT7650
C                                                                       CFFT7660
  914 J = J + 1                                                         CFFT7670
      KK = NP(J)                                                        CFFT7680
      IF (KK .LT. 0) GO TO 914                                          CFFT7690
C                                                                       CFFT7700
      IF (KK .NE. J) GO TO 910                                          CFFT7710
C                                                                       CFFT7720
      NP(J) = -J                                                        CFFT7730
      IF (J .NE. NN) GO TO 914                                          CFFT7740
      MAXF = INC * MAXF                                                 CFFT7750
C                                                                       CFFT7760
C         REORDER A AND B, FOLLOWING PERMUTATION CYCLES.                CFFT7770
C                                                                       CFFT7780
      GO TO 950                                                         CFFT7790
C                                                                       CFFT7800
  924 J = J - 1                                                         CFFT7810
      IF (NP(J) .LT. 0)  GO TO 924                                      CFFT7820
      JJ = JC                                                           CFFT7830
C                                                                       CFFT7840
  926 KSPAN = JJ                                                        CFFT7850
      IF (JJ .GT. MAXF)  KSPAN = MAXF                                   CFFT7860
C                                                                       CFFT7870
      JJ = JJ - KSPAN                                                   CFFT7880
      K = NP(J)                                                         CFFT7890
      KK = JC * K + II + JJ                                             CFFT7900
      K1 = KK + KSPAN                                                   CFFT7910
      K2 = 0                                                            CFFT7920
C                                                                       CFFT7930
  928 K2 = K2 + 1                                                       CFFT7940
      AT(K2) = A(K1)                                                    CFFT7950
      BT(K2) = B(K1)                                                    CFFT7960
C                                                                       CFFT7970
      K1 = K1 - INC                                                     CFFT7980
      IF (K1 .NE. KK) GO TO 928                                         CFFT7990
C                                                                       CFFT8000
  932 K1 = KK + KSPAN                                                   CFFT8010
      K2 = K1 - JC * (K+NP(K))                                          CFFT8020
      K =  -NP(K)                                                       CFFT8030
C                                                                       CFFT8040
  936 A(K1) = A(K2)                                                     CFFT8050
      B(K1) = B(K2)                                                     CFFT8060
C                                                                       CFFT8070
      K1 = K1 - INC                                                     CFFT8080
      K2 = K2 - INC                                                     CFFT8090
      IF (K1 .NE. KK) GO TO 936                                         CFFT8100
```

```
C                                                                       CFFT8110
      KK = K2                                                           CFFT8120
      IF (K .NE.J) GO TO 932                                            CFFT8130
C                                                                       CFFT8140
      K1 = KK + KSPAN                                                   CFFT8150
      K2 = 0                                                            CFFT8160
C                                                                       CFFT8170
  940 K2 = K2 + 1                                                       CFFT8180
      A(K1) = AT(K2)                                                    CFFT8190
      B(K1) = BT(K2)                                                    CFFT8200
C                                                                       CFFT8210
      K1 = K1 + INC                                                     CFFT8220
      IF (K1 .NE. KK) GO TO 940                                         CFFT8230
C                                                                       CFFT8240
      IF (JJ .NE. 0)  GO TO 926                                         CFFT8250
C                                                                       CFFT8260
      IF (J  .NE. 1)  GO TO 924                                         CFFT8270
C                                                                       CFFT8280
  950 J = K3 + 1                                                        CFFT8290
      NT = NT - KSPNN                                                   CFFT8300
      II = NT - INC  + 1                                                CFFT8310
      IF (NT .GE. 0) GO TO 924                                          CFFT8320
C                                                                       CFFT8330
      RETURN                                                            CFFT8340
C                                                                       CFFT8350
C         ERROR TERMINATION: INSUFFICIENT ARRAY STORAGE                 CFFT8360
C                                                                       CFFT8370
  998 ISN = 0                                                           CFFT8380
      STOP 999                                                          CFFT8390
      END                                                               CFFT8400
```

APPENDIX E. MODE1--MODE ONE OPERATION OF VAN HOUTEN'S COMPUTER PROGRAM WITH SOME MODIFICATIONS

1) Program does not use SUBROUTINE COEFF
2) C1 has been corrected to .292857.

```
C      *                                                                    EVE00010
C      *          1) FT02F001 - OUTPUT IN FORM TO BE READ BACK INTO         EVE00020
C      *             PROGRAM UNDER MODE 3 OPERATION (CHANGE MODE=1          EVE00030
C      *             TO MODE=3 ON THE 4TH DATA CARD                         EVE00040
C      *          2) FT08F001 - OUTPUT AS PRINTED UNDER CS                  EVE00050
C      *                                                                    EVE00060
C      *                                                                    EVE00070
C      ********************************************************************EVE00080
C                          MAIN COMPUTER PROGRAM                           EVE00090
C      ********************************************************************EVE00100
C      *                                                                   *EVE00110
C      *                        INTEGRAL INVERSION                         *EVE00120
C      *                                                                   *EVE00130
C      *  REFERENCE, THESIS BY P. E. VAN HOUTEN 'THE APPLICATION OF        *EVE00140
C      *  HOLOGRAPHIC INTERFEROMETRY TO THE DETERMINATION OF DISCON-       *EVE00150
C      *  TINUOUS THREE-DIMENSIONAL DENSITY FIELDS' 14 DEC 72              *EVE00160
C      *        THE PROGRAM CONSIDERS A CIRCULAR REGION OF UNKNOWN         *EVE00170
C      *  REFRACTIVE INDEX FOR WHICH THE TOTAL PHASE SHIFT(PATH LENGTH     *EVE00180
C      *  DIFFERENCE) IS KNOWN FOR NUMEROUS RAYS AT REGULAR INTERVALS      *EVE00190
C      *  FOR A GIVEN VIEWING ANGLE. THE PROGRAM REQUIRES NUMEROUS         *EVE00200
C      *  VIEWS AT REGULARLY SPACED INTERVALS AND A TOTAL ANGULAR          *EVE00210
C      *  COVERAGE OF 180 DEGREES(ASYMMETRIC). THE NECESSARY ANGULAR       *EVE00220
C      *  COVERAGE IS REDUCED BY THE NUMBER OF DEGREES OF SYMMETRY.        *EVE00230
C      *  BY THE USE OF TWO DIMENSIONAL FOURIER TRANSFORM TECHNIQUES       *EVE00240
C      *  IT CAN BE SHOWN THAT  FOR A GIVEN POINT(XO,YO) WITHIN THE        *EVE00250
C      *  CIRCULAR REGION, THE DENSITY F(XO,YO) CAN BE EXPRESSED IN        *EVE00260
C      *  TERMS OF THE PHASE SHIFT INFORMATION G(RC,THETA).                *EVE00270
C      *        IN ORDER TO SAVE COMPUTATION TIME, THE VALUE OF THE        *EVE00280
C      *  INSIDE INTEGRATION IS COMPUTED AT EACH RAY POSITION FOR EACH     *EVE00290
C      *  VIEWING ANGLE. FOR A GIVEN POINT(XO,YO), THE VALUE OF THE        *EVE00300
C      *  INSIDE INTEGRAL IS DETERMINED BY AVERAGING OVER A FINITE         *EVE00310
C      *  PORTION OF A CUBIC POLYNOMIAL FITTED TO THE FOUR CLOSEST         *EVE00320
C      *  VALUES OF THE INSIDE INTEGRAL.  THE INTEGRATION OVER THETA       *EVE00330
C      *  USES THE TRAPEZOIDAL RULE.  THE INTEGRATION OVER RHO USES        *EVE00340
C      *  COTES SIXTH ORDER QUADATURE FORMULA                              *EVE00350
C      *                                                                   *EVE00360
C      *     MEANING OF SYMBOLS USED IN THE PROGRAM                        *EVE00370
C      *                                                                   *EVE00380
C      *  IT(12)  TITLE USED FOR OUTPUT.  FIRST 48 CHARACTERS ON FIRST     *EVE00390
C      *          DATA CARD, FIRST 16 CHARACTERS ON NEXT DATA CARD         *EVE00400
C      *  RHOINF-DENSITY OUTSIDE REGION OF INTEREST IN MG/CC.              *EVE00410
C      *  XLAMDA-WAVELENGTH OF MONOCHROMATIC,COHERENT LIGHT SOURCE IN      *EVE00420
```

```
C      *   MICRONS.                                                          *EVE00430
C      *   RHOINF AND XLAMDA ON THIRD DATA CARD IN (2F10.5) FORMAT           *EVE00440
C      *   MODE     MODE=1  REFRACTIVE INDEX SUPPLIED THROUGH SUBROUTINE     *EVE00450
C      *                    'INPUT'. THE PROGRAM GENERATES FRINGE DATA       *EVE00460
C      *                    AND RECOMPUTES THE REFRACTIVE INDEX FROM         *EVE00470
C      *                    THE GENERATED FRINGE DATA.                       *EVE00480
C      *            MODE=2  REFRACTIVE INDEX SUPPLIED AT GRID POINTS         *EVE00490
C      *                    BY INPUT DATA. PROGRAM PROCEEDS AS IN MODE 1     *EVE00500
C      *            MODE=3 FRINGE DATA SUPPLIED, PROGRAM GENERATES           *EVE00510
C      *                   DENSITY FIELD.                                    *EVE00520
C      *            MODE=4  SAME AS MODE=3 EXCEPT FRINGE DATA IS RECOM-      *EVE00530
C      *                    PUTED AS PER MODE =2.                            *EVE00540
C      *   NP  NUMBER OF EQUALLY SPACED DATA POINTS                          *EVE00550
C      *   NT  NUMBER OF VIEWS SUPPLIED                                      *EVE00560
C      *   KSYM  SYMMETRY OF FIELD  0-4 AXIS OF SYMMETRY                     *EVE00570
C      *                                4 -98  DATA REPEATS EVERY KSYM       *EVE00580
C      *                                       DEGREES.                      *EVE00590
C      *                               99 AXISYMETRIC FIELD                  *EVE00600
C      *   RMAX  RADIUS OF CIRCULAR REGION  IN CENTIMETERS                   *EVE00610
C      *   FILTER = (0.-1.) DETERMINES SPAN OVER WHICH AVERAGING OCCURS      *EVE00620
C      *   THE ABOVE INFORMATION ON THE FOURTH DATA CARD(4I10,2F10.5).       *EVE00630
C      *   X(I)    VECTOR OF GRID POINTS ALONG THETA = 0 DEGREES             *EVE00640
C      *           X(1)=-RMAX     X(NP)=+RMAX                                *EVE00650
C      *   Y(J)    VECTOR OF GRID POINTS ALONG THETA=90 DEGREES              *EVE00660
C      *           Y(1)=-RMAX   Y(NP)=+RMAX                                  *EVE00670
C      *   A(I,J)    DENSITY OR REFRACTIVE INDEX AT THE POINT X(I),Y(J).     *EVE00680
C      *   RB(I) VECTOR OF POINTS ON THE HOLOGRAPHIC PLATE RB(1)=-2 RMAX     *EVE00690
C      *   RB(2 NP)=2 RMAX.                                                  *EVE00700
C      *   DR    SPACING OF GRID,SPACING OF RAYS.  DR=2*RMAX/(NP-1)          *EVE00710
C      *   TH(K)    VECTOR OF VIEWING ANGLES  IN DEGREES                     *EVE00720
C      *   B(I,K)    FRINGE SHIFT INFORMATION AT THE POINT(RB(I),TH(K)).     *EVE00730
C      *   B(I,K) IS COPIED INTO BB(I,K). THE VALUE OF THE INSIDE            *EVE00740
C      *   INTEGRATION IS THEN STORED IN B(I,K).                             *EVE00750
C      *   DASYM(K) SYMMETRY OF DATA(B(I,K). 0 NO SYMETRY                    *EVE00760
C      *                                     1 DATA SYMETRIC ABOUT RB=0.     *EVE00770
C      *   IF DASYM = 1.0 ONLY (NP+1)/2 DATA POINTS SHOULD BE FURNISHED.     *EVE00780
C      *   BD(L) IS A WORK VECTOR USED BY THE PROGRAM.                       *EVE00790
C      *   FOR MODE=1 OR  MODE=2 THE FIFTH DATA CARD IS THE VECTOR           *EVE00800
C      *   TH(K) IN (8F10.5) FORMAT                                          *EVE00810
C      *   FOR MODE 2 OPERATION THE REFRACTIVE INDEX FOLLOWS IN 8F10.5       *EVE00820
C      *   FORMAT.                                                           *EVE00830
C      *   FOR MODE=3 AND MODE=4 OPERATION, THE FIFTH DATA CARD IS           *EVE00840
C      *   TH(1) AND DASYM(1) IN (2F10.5) FORMAT.  THIS IS FOLLOWED          *EVE00850
C      *   BY THE FRINGE INFORMATION FOR THAT VIEW (B(I,1) ETC UNTIL         *EVE00860
C      *   ALL VIEWS AND FRINGE SHIFT HAS BEEN SUPPLIED                      *EVE00870
C      *   IF AXIS OF SYMMETRY EXISTS,THETA=0. MUST CORRESPOND TO ONE OF     *EVE00880
C      *   THE AXIS OF SYMMETRY.                                             *EVE00890
C      *                                                                     *EVE00900
```

```
C     ****************************************************************EVE00910
      REAL*8 IT(12),LAB,ITSUB(12)/'REFRACTI','VE INDEX','DENSITY ','IN MEVE00920
     1G/CC','    THET','A=  0DEG','    THET','A= 45DEG','    THET','A=-4EVE00930
     15DEG'/                                                           EVE00940
      DIMENSION X(151),Y(151),A(151,151),B(151,20),BD(811),TH(36),     EVE00950
     1BB(301,20),RB(301),DASYM(20),SS(36),CC(36)                       EVE00960
      INTEGER CLOCK(6)                                                 EVE00970
      DATA C1,C2,C3,C4/.292857,1.54285,.192857,1.94285/,LAB/'        '/ EVE00980
  100 FORMAT (4I10,2F10.5)                                             EVE00990
  101 FORMAT('0',T20,'   MODE     NUM PTS  NUM VIEWS   KSYM   RADIUS    EVE01000
     1GRID SPACING  FILTER')                                           EVE01010
  102 FORMAT('0',T17,4I10,3F10.5)                                      EVE01020
  103 FORMAT(8F10.5)                                                   EVE01030
  104 FORMAT('0',T7,20F6.1//)                                          EVE01040
  105 FORMAT('0',F5.2,20F6.2)                                          EVE01050
  106 FORMAT('0',//)                                                   EVE01060
  107 FORMAT('0',T9,'MODE 1',T26,'MODE 3',T45,'VALUE'/,T9,'GIVEN    COMPEVE01070
     1    DENSITY',T47,'OF'/,'RADIUS  REFRAC   REFRAC  FIELD     FRINGE EVE01080
     2  INSIDE'/,'  CM    INDEX    INDEX   MG/CC     SHIFT    INTEGRAL' EVE01090
     3//)                                                              EVE01100
  108 FORMAT('0',T20,'RHOINF =',F7.5,' MG/CC  LAMDA=',F7.5,' MICRONS')  EVE01110
  110 FORMAT('1',T7,21F6.2//)                                          EVE01120
  111 FORMAT('0',F5.2,1X,21F6.2)                                       EVE01130
  112 FORMAT(6A8)                                                      EVE01140
  113 FORMAT(' ',T40,'Y AXIS(THETA=90 DEGREES) VERTICAL'/,T40,         EVE01150
     1'X AXIS(THETA = 0 DEGREES) HORIZONTAL'//)                        EVE01160
  114 FORMAT('0',T40,'DENSITY FIELD IN MG/CC FOLLOWS')                 EVE01170
  115 FORMAT('0',T40,'INDEX OF REFRACTION FOLLOWS')                    EVE01180
  116 FORMAT('1','THE EXPANDED FRINGE SHIFT INFORMATION IS AS FOLLOWS')EVE01190
  117 FORMAT('1','THE VALUE OF THE INSIDE INTEGRATION IS AS FOLLOWS'//) EVE01200
  118 FORMAT(10F7.4)                                                   EVE01210
      READ(1,112) (IT(I),I=1,12)                                       EVE01220
      READ(1,103) RHOINF,XLAMD                                         EVE01230
      READ(1,100) MODE,NP,NT,KSYM,RMAX,FILTER                          EVE01240
      WRITE(8,108) RHOINF,XLAMD                                        EVE01250
      FNP=FLOAT(NP)                                                    EVE01260
      DR=2.*RMAX/(FNP-1)                                               EVE01270
      WRITE(8,101)                                                     EVE01280
      WRITE(8,102) MODE,NP,NT,KSYM,RMAX,DR,FILTER                      EVE01290
      N=(NP+1)/2                                                       EVE01300
      NPL=(NP-1)/2                                                     EVE01310
      NPP=2*NP-1                                                       EVE01320
      MAX=NP+NPL                                                       EVE01330
      RM2=RMAX**2                                                      EVE01340
      NP5=3*NP-2                                                       EVE01350
      FNP5=FLOAT(NP5)                                                  EVE01360
      NPM1=NP-1                                                        EVE01370
      FT2=FILTER*FILTER                                                EVE01380
```

```
C                                                                       EVE01390
C*****ZERO MATRICES AND SET GRID ************************************** EVE01400
C                                                                       EVE01410
      DO 2 I=1,NP                                                       EVE01420
      DO 1 J=1,NP                                                       EVE01430
    1 A(I,J)=0.0                                                        EVE01440
      DO 2 K=1,NT                                                       EVE01450
    2 B(I,K)=0.0                                                        EVE01460
      X(1)=-RMAX                                                        EVE01470
      Y(1)=X(1)                                                         EVE01480
      DO 3 I=2,NP                                                       EVE01490
      X(I)=X(I-1)+DR                                                    EVE01500
    3 Y(I)=X(I)                                                         EVE01510
C*****READ VIEWING ANGLES AND SET SIN AND COSINE FOR MODE=1,2********** EVE01520
C                                                                       EVE01530
    4 READ(1,103) (TH(K),K=1,NT)                                        EVE01540
      DO 5 K=1,NT                                                       EVE01550
      THETA=TH(K)/57.296                                                EVE01560
      CC(K)=COS(THETA)                                                  EVE01570
    5 SS(K)=SIN(THETA)                                                  EVE01580
    7 IF(KSYM.EQ.99) GO TO 13                                           EVE01590
C                                                                       EVE01600
C*****SET REFRACTIVE INDEX AND WRITE SAME FOR MODE ONE OPERATION******* EVE01610
C                                                                       EVE01620
      DO 8 I=1,NP                                                       EVE01630
      DO 8 J=1,NP                                                       EVE01640
      XI=X(I)                                                           EVE01650
      YJ=Y(J)                                                           EVE01660
      CALL INPUT(XI,YJ,VAL)                                             EVE01670
    8 A(I,J)=VAL                                                        EVE01680
    9 WRITE(8,115)                                                      EVE01690
      WRITE(8,113)                                                      EVE01700
      K=1                                                               EVE01710
      K21=21                                                            EVE01720
   11 WRITE(8,106)                                                      EVE01730
      WRITE(8,110) (X(I),I=K,K21)                                       EVE01740
      DO 12 I=1,NP                                                      EVE01750
      NN=NP-I+1                                                         EVE01760
   12 WRITE(8,111) Y(NN),(A(J,NN),J=K,K21)                              EVE01770
      IF(K21.EQ.NP) GO TO 15                                            EVE01780
      K=K21+1                                                           EVE01790
      K21=K21+21                                                        EVE01800
      IF(K21.GT.NP) K21=NP                                              EVE01810
      GO TO 11                                                          EVE01820
C*********************************************************************** EVE01830
C                                                                       EVE01840
C*****SET REFRACTIVE INDEX FOR AXISYMETRIC CASE************************ EVE01850
C                                                                       EVE01860
```

```
   13 DO 14 I=1,NP                                                          EVE01870
   14 CALL INPUT(X(I),0,A(I,N))                                             EVE01880
   15 CONTINUE                                                              EVE01890
C                                                                           EVE01900
C***************************************************************************EVE01910
C                                                                           EVE01920
C                                                                           EVE01930
C*****INTEGRATE FOR FINGE SHIFT FOR MODE ONE CASE************************** EVE01940
C                                                                           EVE01950
      DO 18 K=1,NT                                                          EVE01960
      R=-RMAX                                                               EVE01970
      B(1,K)=0.0                                                            EVE01980
      B(NP,K)=0.0                                                           EVE01990
      DO 18 I=2,NPM1                                                        EVE02000
      R=R+DR                                                                EVE02010
      P1=SQRT(RM2-R**2)                                                     EVE02020
      DP=2.*P1/FNP5                                                         EVE02030
      P=-P1                                                                 EVE02040
      DO 16 J=1,NP5                                                         EVE02050
      YJ=R*CC(K)+P*SS(K)                                                    EVE02060
      XI=-R*SS(K)+P*CC(K)                                                   EVE02070
      CALL INPUT(XI,YJ,VAL)                                                 EVE02080
      P=P+DP                                                                EVE02090
   16 BD(J)=VAL                                                             EVE02100
      SUM=0.                                                                EVE02110
      NPP2=NP5-2                                                            EVE02120
      DO 17 J=1,NPP2,2                                                      EVE02130
C*****INTEGRATION USES SIMPSON'S RULE ***********************               EVE02140
   17 SUM=SUM+BD(J)+4.*BD(J+1)+BD(J+2)                                      EVE02150
   18 B(I,K)=SUM*DP/3.                                                      EVE02160
      GO TO 34                                                              EVE02170
C                                                                           EVE02180
C*****SET RADIUS VECTOR AND EXTEND FRINGE INFORMATION TO TWICE RMAX****     EVE02190
C                                                                           EVE02200
   34 RB(1)=-2.*RMAX                                                        EVE02210
      DO 35 I=2,NPP                                                         EVE02220
   35 RB(I)=RB(I-1)+DR                                                      EVE02230
      DO 38 I=1,NT                                                          EVE02240
      Q1=B(1,I)                                                             EVE02250
      C2=(B(2,I)-B(1,I))/DR                                                 EVE02260
      DO 36 J=1,NPL                                                         EVE02270
   36 BB(J,I)=3.*Q2+RB(J)*(3.*Q2-4.*Q1/3.+RB(J)*(C2/1.5-C1/2.25))           EVE02280
      DO 37 K=1,NP                                                          EVE02290
   37 BB(K+NPL,I)=B(K,I)                                                    EVE02300
      Q1=B(NP,I)                                                            EVE02310
      Q2=(B(NP,I)-B(NP-1,I))/DR                                             EVE02320
      L=NP+NPL                                                              EVE02330
      DO 38 K=L,NPP                                                         EVE02340
```

```
   38 BB(K,I)=-3.*Q2+RB(K)*(3.*Q2+4.*Q1/3.+RB(K)*(-Q2/1.5-Q1/2.25))      EVE02350
      DO 381 I=1,NT                                                      EVE02360
      DO 382 J=1,NPP                                                     EVE02370
      WRITE(3,1001) BB(J,I),J,I                                          EVE02380
 1001 FORMAT(F10.5,2I10)                                                 EVE02390
  382 CONTINUE                                                           EVE02400
  381 CONTINUE                                                           EVE02410
C                                                                        EVE02420
C*************************************************************         EVE02430
C                                                                        EVE02440
C                                                                        EVE02450
C*****CALCULATE VALUE OF INSIDE INTEGRAL ***************************    EVE02460
C                                                                        EVE02470
      DO 43 I=1,NT                                                       EVE02500
      DO 43 J=1,NP                                                       EVE02510
      M=NPL+J                                                            EVE02520
      BD(1)=0.                                                           EVE02530
      DO 39 K=1,MAX                                                      EVE02540
      MKP=M+K                                                            EVE02550
      MKM=M-K                                                            EVE02560
      IF(MKP.GT.NPP) MKP=NPP                                             EVE02570
      IF(MKM.LT.1) MKM=1                                                 EVE02580
      K1=K**2                                                            EVE02590
   39 BD(K+1)=(BB(MKP,I)+BB(MKM,I)-2.*BB(M,I))/(FLOAT(K1)*DR)            EVE02600
      IF(J.GT.20) GO TO 391                                              EVE02610
      WRITE(3,1002) (BD(L),L=1,98)                                       EVE02620
 1002 FORMAT(10F8.3)                                                     EVE02630
  391 CONTINUE                                                           EVE02640
                                                                         EVE02650
      VAL=0.                                                             EVE02660
      K1=1                                                               EVE02670
   40 K2=K1+6                                                            EVE02680
      IF(K2.GT.MAX) GO TO 41                                             EVE02690
C     CCTES SIXTH ORDER QUADATURE FORMULA*****************               EVE02700
      VAL=VAL+C1*(BD(K1)+BD(K2))+C2*(BD(K1+1)+BD(K2-1))+C3*(BD(K1+2)+BD(EVE02710
     1K2-2))+C4*BD(K1+3)                                                 EVE02720
      K1=K2                                                              EVE02730
      GO TO 40                                                           EVE02740
   41 SUM=0.                                                             EVE02750
C     CONTRIBUTION OF LAST TERMS                                         EVE02760
      DO 42 K=K1,MAX                                                     EVE02770
   42 SUM=SUM+BD(K)+BD(K+1)                                              EVE02780
C     SUM COTES PLUS LAST TERMS PLUS INTEGRATION FROM 2 RMAX TO INFINITYEVE02790
      B(J,I)= VAL+SUM/2.-.666667*BB(M,I)/RMAX                            EVE02800
      WRITE(3,1001) B(J,I),J,I                                           EVE02810
   43 CONTINUE                                                           EVE02820
      IF(KSYM.LT.99) GO TO 48                                            EVE02830
C                                                                        EVE02840
```

```
C*****************************************************************************EVE02850
C                                                                               EVE02860
C                                                                               EVE02870
C*****INTEGRATE OVER THETA FOR AXISYMETRIC CASE************************         EVE02880
C                                                                               EVE02890
      DEL=-.004421                                                              EVE02900
      M=1                                                                       EVE02910
      TH(1)=0.0                                                                 EVE02920
      SS(1)=0.0                                                                 EVE02930
      DO 44 I=2,19                                                              EVE02940
      TH(I)=TH(I-1)+5.                                                          EVE02950
   44 SS(I)=SIN(TH(I)/57.296)                                                   EVE02960
      WRITE(8,107)                                                              EVE02970
      DO 47 I=1,NP                                                              EVE02980
      DO 45 K=1,19                                                              EVE02990
      R=-X(I)*SS(K)                                                             EVE03000
      L=(R+RMAX+DR)/DR                                                          EVE03010
      IF(L.GE.NP) GO TO 460                                                     EVE03020
      IF(L.LT.1) GO TO 460                                                      EVE03030
      LP2=L+2                                                                   EVE03040
      LP1=L+1                                                                   EVE03050
      LM1=L-1                                                                   EVE03060
      IF(LP2.GT.NP) LP2=NP                                                      EVE03070
      IF(LM1.LT.1) LM1=1                                                        EVE03080
      P=(R-X(L))/DR                                                             EVE03090
      COF3=.1666666*(B(LP2,K)-B(LM1,K))-.5*(B(LP1,K)-B(L,K))                    EVE03100
      COF2=.5*(B(LP1,K)+B(LM1,K))-B(L,K)                                        EVE03110
      COF1=-.1666666*B(LP2,K)+B(LP1,K)-.333333*B(LM1,K)-.5*B(L,K)               EVE03120
      P2=P*P                                                                    EVE03130
C*****AMOUNT OF FILTERING DEPENDS ON FT2=FILTER*FILTER**********                EVE03140
   45 BD(K)=P*(P2+FT2)*COF3+(P2+.666666*FT2)*COF2+P*COF1+B(L,K)                 EVE03150
      SUM=0.0                                                                   EVE03160
C     TRAPEZOIDAL INTEGRATION ON THETA   ******************                     EVE03170
      DO 46 K=2,19                                                              EVE03180
      SUM=SUM+BD(K)+BD(K-1)                                                     EVE03190
   46 CONTINUE                                                                  EVE03200
      A(I,1)=SUM*DEL                                                            EVE03210
  46C CONTINUE                                                                  EVE03220
C     OUTPUT IS DENSITY FIELD IF MODE=3*********************                    EVE03230
      IF(MODE.LT.3) GO TO 47                                                    EVE03240
      A(I,2)=RHOINF+.4443*XLAMD*A(I,1)                                          EVE03250
   47 WRITE(8,105) X(I),A(I,N),A(I,1),A(I,2),BB(NPL+I,1),B(I,1)                 EVE03260
                                                                                EVE03270
C*****************************************************************************EVE03280
C                                                                               EVE03290
C                                                                               EVE03300
C*****PLOT AXISYMETRIC DENSITY FIELD***********************************         EVE03310
C                                                                               EVE03320
```

```
C      REFRACTIVE INDEX SUPPLIED*****************************                EVE03330
       GC TO 75                                                              EVE03340
   48 CCNTINUE                                                               EVE03350
       M=NT+1                                                                EVE03380
       IF(KSYM.EQ.0) GO TO 51                                                EVE03390
       IF(KSYM.GT.4) GO TO 53                                                EVE03400
C                                                                            EVE03410
C*****GENERATE ADDITIONAL DATA DEPENDING ON SYMETRY OF FIELD***********      EVE03420
C                                                                            EVE03430
       DT=TH(2)-TH(1)                                                        EVE03440
       NTP=NT                                                                EVE03450
       DO 50 J=1,KSYM                                                        EVE03460
       K=J*NT                                                                EVE03470
       DO 49 I=1,NTP                                                         EVE03480
       KK=K+I                                                                EVE03490
       TH(KK)=TH(KK-1)+DT                                                    EVE03500
       KM=K-I+1                                                              EVE03510
       DO 49 L=1,NP                                                          EVE03520
   49 B(L,KK)=B(L,KM)                                                        EVE03530
   50 NTP=2*NTP                                                              EVE03540
       M=KK+1                                                                EVE03550
C                                                                            EVE03560
C*************************************************************************  EVE03570
C*****GENERATE DATA FOR 180 DEGREE VIEW*********************************      EVE03580
C                                                                            EVE03590
   51 TH(M)=TH(1)+180.                                                       EVE03600
       DO 52 I=1,NP                                                          EVE03610
       K=(NP-I+1)                                                            EVE03620
   52 B(K,M)=B(I,1)                                                          EVE03630
C                                                                            EVE03640
C*************************************************************************  EVE03650
C                                                                            EVE03660
       GO TO 55                                                              EVE03670
C                                                                            EVE03680
C*****GENERATE ADDITIONAL DATA FOR REPEATING DATA**********************      EVE03690
C                                                                            EVE03700
   53 M=180/KSYM+1                                                           EVE03710
       DO 54 I=2,M                                                           EVE03720
       TH(I)=TH(I-1)+FLOAT(KSYM)                                             EVE03730
       DO 54 J=1,NP                                                          EVE03740
   54 B(J,I)=B(J,I-1)                                                        EVE03750
   55 CONTINUE                                                               EVE03760
C                                                                            EVE03770
C*************************************************************************  EVE03780
C                                                                            EVE03790
C                                                                            EVE03800
C*****INTEGRATE ON THETA FOR ASYMETRIC FIELD***************************      EVE03810
C                                                                            EVE03820
```

```
      MC=M-1                                                                EVE03830
      DEL=-1./(2.*3.1416*FLOAT(MD))                                         EVE03840
      DO 56 I=1,M                                                           EVE03850
      CC(I)=COS(TH(I)/57.296)                                               EVE03860
   56 SS(I)=SIN(TH(I)/57.296)                                               EVE03870
      DO 61 LL= 1,NP                                                        EVE03880
      J=NP+ 1-LL                                                            EVE03890
      DO 60 I= 1,NP                                                         EVE03900
      A(I,J)=0.                                                             EVE03910
      DO 58 K=1,M                                                           EVE03920
      R=Y(J)*CC(K)-X(I)*SS(K)                                               EVE03930
      NN=(R+RMAX+DR)/DR                                                     EVE03940
      IF(NN.GE.NP) GO TO 60                                                 EVE03950
      IF(NN.LT.1) GO TO 60                                                  EVE03960
      LP1=NN+1                                                              EVE03970
      LM1=NN-1                                                              EVE03980
      IF(LP2.GT.NP) LP2=NP                                                  EVE03990
      IF(LM1.LT.1) LM1=1                                                    EVE04000
      P=(R-X(NN))/DR                                                        EVE04010
      COF3=.1666666*(B(LP2,K)-B(LM1,K))-.5*(B(LP1,K)-B(NN,K))               EVE04020
      COF2=.5*(B(LP1,K)+B(LM1,K))-B(NN,K)                                   EVE04030
      COF1=-.1666666*B(LP2,K)+B(LP1,K)-.333333*B(LM1,K)-.5*B(NN,K)          EVE04040
      P2=P*P                                                                EVE04050
C*****AMOUNT OF FILTERING DEPENDS ON FT2=FILTER*FILTER**********            EVE04060
   58 BD(K)=P*(P2+FT2)*COF3+(P2+.666666*FT2)*COF2+P*COF1+B(NN,K)             EVE04070
C     INTEGRATE OVER THETA USING TRAPEZOIDAL INTEGRATION*********           EVE04080
      VAL=0.0                                                               EVE04090
      DO 59 L=2,M                                                           EVE04100
   59 VAL=VAL+BD(L)+BD(L-1)                                                 EVE04110
      A(I,J)=VAL*DEL/2.                                                     EVE04120
C*************************************************************            EVE04130
   60 CONTINUE                                                              EVE04140
   61 CONTINUE                                                              EVE04150
C                                                                           EVE04160
C***********************************************************************   EVE04170
C                                                                           EVE04180
      NPN=NP+NPL                                                            EVE04190
      IF(MODE.LT.4) GO TO 63                                                EVE04200
   63 IF(MODE.LT.3) GO TO 65                                                EVE04210
C                                                                           EVE04220
C*****CHANGE REFRACTIVE INDEX FIELD TO DENSITY FIELD*********************   EVE04230
C                                                                           EVE04240
      DO 64 I=1,NP                                                          EVE04250
      DO 64 J=1,NP                                                          EVE04260
   64 A(I,J)=RHOINF+.4443*XLAMD*A(I,J)                                      EVE04270
C                                                                           EVE04280
C***********************************************************************   EVE04290
C                                                                           EVE04300
```

```
      WRITE(8,114)                                                      EVE04310
      GO TO 66                                                          EVE04320
   65 WRITE(8,115)                                                      EVE04330
   66 WRITE(8,113)                                                      EVE04340
C                                                                       EVE04350
C*****WRITE REFRACTIVE INDEX OR DENSITY FIELD*************************  EVE04360
C                                                                       EVE04370
      K=1                                                               EVE04380
      K21=21                                                            EVE04390
   67 WRITE(8,106)                                                      EVE04400
      WRITE(8,110) (X(I),I=K,K21)                                       EVE04410
      DO 68 I= 1,NP                                                     EVE04420
      NN=NP-I+1                                                         EVE04430
   68 WRITE(8,111) Y(NN),(A(J,NN),J=K,K21)                              EVE04440
      IF(K21.EQ.NP) GO TO 69                                            EVE04450
      K=K21+1                                                           EVE04460
      K21=K21+21                                                        EVE04470
      IF(K21.GT.NP) K21=NP                                              EVE04480
      GO TO 67                                                          EVE04490
C*********************************************************************  EVE04500
C                                                                       EVE04510
   69 WRITE(8,116)                                                      EVE04520
C                                                                       EVE04530
C*****WRITE FRINGE SHIFT INFORMATION**********************************  EVE04540
C                                                                       EVE04550
      WRITE(8,104) (TH(K),K=1,NT)                                       EVE04560
      DO 70 I=N,NPN                                                     EVE04570
   70 WRITE(8,105) RB(I),(BB(I,J),J=1,NT)                               EVE04580
C                                                                       EVE04590
      WRITE(8,116)                                                      EVE04600
      WRITE(8,104) (TH(K),K=1,NT)                                       EVE04610
      DO 700 J=1,NT                                                     EVE04620
      WRITE(8,103) TH(J)                                                EVE04630
      WRITE(8,103)(BB(I,J),I=N,NPN)                                     EVE04640
  700 CONTINUE                                                          EVE04650
C                                                                       EVE04660
                                                                        EVE04670
      WRITE(2,112)(IT(I),I=1,12)                                        EVE04680
      WRITE(2,103) RHOINF,XLAMD                                         EVE04690
      WRITE(2,100) MODE,NP,NT,KSYM,RMAX,FILTER                          EVE04700
      DO 710 J=1,NT                                                     EVE04710
      WRITE(2,103) TH(J)                                                EVE04720
      WRITE(2,103)(BB(I,J),I=N,NPN)                                     EVE04730
  710 CONTINUE                                                          EVE04740
C                                                                       EVE04750
      WRITE(8,117)                                                      EVE04760
      WRITE(8,104) (TH(K),K=1,M)                                        EVE04770
      DO 71 I=1,NP                                                      EVE04780
```

```
   71 WRITE(8,105) X(I),(B(I,J),J=1,M)                                  EVE04790
   75 CONTINUE                                                          EVE04800
      STOP                                                              EVE04810
      END                                                               EVE04820
      SUBROUTINE COEF(VAL,SLOPE,U,CU2,CU,CONST)                         EVE04830
C     ***************************************************************EVE04840
C     *                                                             *EVE04850
C     *  SUBROUTINE COEF CALCULATES THE COEFICIENTS USED BY  THE    *EVE04860
C     *  FUNCTION (F1) TO EXPAND THE FRINGE INFORMATION TO 2 RMAX   *EVE04870
C     *                                                             *EVE04880
C     ***************************************************************EVE04890
      U2=U*U                                                            EVE04900
      CU2=-SLOPE/U-VAL/U2                                               EVE04910
      CU=-VAL/U-3.*U*CU2                                                EVE04920
      CONST=VAL-CU*U-CU2*U2                                             EVE04930
      RETURN                                                            EVE04940
      END                                                               EVE04950
      SUBROUTINE INPUT(XI,YJ,VAL)                                       EVE04960
C     ***************************************************************EVE04970
C     *                                                             *EVE04980
C     *  SUBROUTINE INPUT IS SUPPLIED BY THE USER FOR MODE I        *EVE04990
C     *  OPERATION.  THE MAIN PROGRAM PASSES THE POINT (XI,YJ) AND  *EVE05000
C     *  'INPUT' SUPPLIES THE VALUE OF THE REFRACTIVE INDEX AT THAT *EVE05010
C     *  POINT USING ANY ARITHMETIC OR LOGIC STATMENT NECESSARY.    *EVE05020
C     *                                                             *EVE05030
      Y2=YJ*YJ                                                          EVE05040
      X2=XI*XI                                                          EVE05050
      VAL=(1.-Y2)*EXP(-2.*X2)                                           EVE05060
      R2=Y2+X2                                                          EVE05070
      IF(R2.GT.1.5) VAL=0.                                              EVE05080
      RETURN                                                            EVE05090
      END                                                               EVE05100
```

APPENDIX F. ALTERNATE PROGRAMS DESIGNED TO CALCULATE THE INSIDE INTEGRAL

APPENDIX F.1 JONES1--USING CONVOLUTION VALUES OBTAINED NUMERICALLY

See Appendix F.2 for expanded fringe shift information used as input fringe data for $-1.5 \leq R \leq 1.5$

```
      DIMENSION X(513),Y(513),Z(513) ,XD(513),YD(513),B(513)          HAL00010
      N=512                                                           HAL00020
      PY=3.141593                                                     HAL00030
      R=12.0                                                          HAL00040
      FN=N                                                            HAL00050
      DR=2.*R/FN                                                      HAL00060
C****LIMITS FOR READING DATA AND PRINTING OUTPUTS*****************    HAL00070
      INTB=1                                                          HAL00080
      INTE=65                                                         HAL00090
      INTPRB=256                                                      HAL00100
      INTPRE=322                                                      HAL00110
      NP1=N+1                                                         HAL00120
      DO 21 I=1,NP1                                                   HAL00130
      Y(I)=0.0                                                        HAL00140
   21 CONTINUE                                                        HAL00150
      READ (1,301) (Y(M),M=INTB,INTE)                                 HAL00160
  301 FORMAT(F10.5)                                                   HAL00170
      DO 22 I=1,NP1                                                   HAL00180
      B(I)=0.0                                                        HAL00190
      IF (I.LE.INTPRB) GO TO 22                                       HAL00200
      IF (I.GE.INTPRE) GO TO 22                                       HAL00210
C****PRINT CHECK FOR PROPER INPUT OF FRINGE DATA*****************    HAL00220
      WRITE (4,101) Y(I),B(I),I                                       HAL00230
   22 CONTINUE                                                        HAL00240
                                                                      HAL00250
      DO 5 K=1,NP1                                                    HAL00260
      C=-R+DR*(K-1)                                                   HAL00270
      IF(C.EQ.0.0) GO TO 6                                            HAL00280
      X(K)=1./C                                                       HAL00290
      GO TO 7                                                         HAL00300
    6 X(K)=0.                                                         HAL00310
    7 CONTINUE                                                        HAL00320
      IF (K.LE.INTPRB) GO TO 5                                        HAL00330
      IF (K.GE.INTPRE) GO TO 5                                        HAL00340
C****PRINT CHECK FOR PROPER INPUT OF 1/R VECTOR*****************     HAL00350
      WRITE(4,201) X(K),Y(K),K,C                                      HAL00360
  201 FORMAT(2F10.5,I10,F10.5)                                        HAL00370
    5 CONTINUE                                                        HAL00380
      DO 10 I=1,N                                                     HAL00390
      Z(I)=0.0                                                        HAL00400
      DO 9 J=1,I                                                      HAL00410
      Z(I)=Z(I)+X(J)*Y(I+1-J)                                         HAL00420
```

```
    9 CONTINUE                                                          HAL00430
      Z(I)=Z(I)*DR                                                      HAL00440
      IF (I.LE.INTPRB) GO TO 10                                         HAL00450
      IF (I.GE.INTPRE) GO TO 10                                         HAL00460
C****PRINT CHECK OF VALUE OF NUMERICAL CONVOLUTION*************         HAL00470
      WRITE(4,101) Y(I),Z(I),I                                          HAL00480
  101 FORMAT(2F10.5,I10)                                                HAL00490
   10 CONTINUE                                                          HAL00500
      CALL CFFT2(Z,B,N,N,N,-1)                                          HAL00510
      DO 15 I=1,N                                                       HAL00520
      Z(I)=Z(I)/FN                                                      HAL00530
      B(I)=B(I)/FN                                                      HAL00540
      T=2.*PY*(I-1.)/FN                                                 HAL00550
      XD(I)=-SIN(T)*B(I)/DR                                             HAL00560
      YD(I)=SIN(T)*Z(I)/DR                                              HAL00570
      IF (I.LE.INTPRB) GO TO 15                                         HAL00580
      IF (I.GE.INTPRE) GO TO 15                                         HAL00590
C****PRINT CHECK OF FOURIER COEFFICIENTS OF CONVOLUTION AND THEIR       HAL00600
C****DERIVATIVES.***************************************************   HAL00610
      WRITE (4,501) Z(I),B(I),I,XD(I),YD(I)                             HAL00620
  501 FORMAT(2F10.5,I10,2F10.5)                                         HAL00630
   15 CONTINUE                                                          HAL00640
      CALL CFFT2(XD,YD,N,N,N,1)                                         HAL00650
      DO 30 J=1,N                                                       HAL00660
      WRITE (4,101)XD(J),YD(J),J                                        HAL00670
C****REAL AND IMAGINARY COEFFICIENTS OF THE INSIDE                      HAL00680
C****INTEGRAL VALUES.***********************************************   HAL00690
   30 CONTINUE                                                          HAL00700
      DO 35 I=257,321                                                   HAL00710
C****CREATING A SEPARATE FILE OF THE REAL VALUES OF THE INSIDE INTEGRAL.HAL00720
      WRITE (3,601) XD(I)                                               HAL00730
  601 FORMAT (F10.5)                                                    HAL00740
   35 CONTINUE                                                          HAL00750
      STOP                                                              HAL00760
      END                                                               HAL00770
```

APPENDIX F.2. JONES2--USING CONVOLUTION VALUES OBTAINED WITH FFT METHODS

See Appendix F.3 for expanded fringe shift information used as input fringe data for

$1.5 \leq R \leq 1.5$

```
      DIMENSION A(513) ,B(513),D(513),FA(513),FB(513),FX(513)                HAC00010
      DIMENSION FY(513),XC(513),YC(513),X(513),Y(513),XD(513),YD(513)        HAC00020
      PY=3.141592653                                                         HAC00030
      R=12.0                                                                 HAC00040
      N=512                                                                  HAC00050
      FN=N                                                                   HAC00060
      DR=2.*R/FN                                                             HAC00070
      INTB=1                                                                 HAC00080
      INTE=65                                                                HAC00090
      INTPRB=256                                                             HAC00100
      INTPRE=322                                                             HAC00110
      NP1=N+1                                                                HAC00120
C  INITIALIZE VECTORS TO ZERO                                                HAC00130
      DO 110 I=1,NP1                                                         HAC00140
      A(I)=0.0                                                               HAC00150
      B(I)=0.0                                                               HAC00160
      X(I)=0.0                                                               HAC00170
      Y(I)=0.0                                                               HAC00180
  110 CONTINUE                                                               HAC00190
C  FRINGE DATA STORED IN A(I) FOR I FROM INTB-INTE.                          HAC00200
      DO 120 I=INTB,INTE                                                     HAC00210
      READ (1,1001) A(I)                                                     HAC00220
 1001 FORMAT (F10.5)                                                         HAC00230
  120 CONTINUE                                                               HAC00240
C  1/Z STORED IN X(I) FOR I FROM 1,N.                                        HAC00250
      DO 130 I=1,N                                                           HAC00260
      C=-R+DR*(I-1)                                                          HAC00270
      IF(C.EQ.0.00) GO TO 9                                                  HAC00280
      X(I)=1./C                                                              HAC00290
      GO TO 11                                                               HAC00300
    9 X(I)=0.0                                                               HAC00310
   11 CONTINUE                                                               HAC00320
      IF (I.LE.INTPRB) GO TO 130                                             HAC00330
      IF (I.GE.INTPRE) GO TO 130                                             HAC00340
C  PRINTOUT CHECK TO ENSURE DATA INPUTTED CORRECTLY.                         HAC00350
      WRITE (4,101) X(I),Y(I),I,A(I),B(I),C                                  HAC00360
  101 FORMAT (2F10.5,I10,3F10.5)                                             HAC00370
  130 CONTINUE                                                               HAC00380
C  TAKING FFT OF INPUT DATA .                                                HAC00390
C  REAL COEFFICIENTS RETURNED AND PLACED IN FA AND FX.                       HAC00400
C  IMAGINARY COEFFICIENTS RETURNED AND STORED IN FB AND FY.                  HAC00410
      CALL CFFT2(X,Y,N,N,N,-1)                                               HAC00420
```

```
      CALL CFFT2(A,B,N,N,N,-1)                                          HAC00430
      DO 20 J=1,N                                                       HAC00440
      FX(J)=X(J)/FN                                                     HAC00450
      FY(J)=Y(J)/FN                                                     HAC00460
      FA(J)=A(J)/FN                                                     HAC00470
      FB(J)=B(J)/FN                                                     HAC00480
      IF (J.LE.INTPRB) GO TO 20                                         HAC00490
      IF (J.GE.INTPRE) GO TO 20                                         HAC00500
C  PRINT CHECK OF FOURIER COEFFICIENTS OF FRINGE DATA AND 1/C.          HAC00510
      WRITE (4,201) FX(J),FY(J),J,FA(J),FB(J)                           HAC00520
  201 FORMAT (2F10.5,I10,2F10.5)                                        HAC00530
   20 CONTINUE                                                          HAC00540
      DO 21 J=1,N                                                       HAC00550
      XC(J)=FX(J)*FA(J)-FY(J)*FB(J)                                     HAC00560
      YC(J)=FX(J)*FB(J)+FY(J)*FA(J)                                     HAC00570
      XC(J)=XC(J)*2.*R                                                  HAC00580
      YC(J)=YC(J)*2.*R                                                  HAC00590
      IF (J.LE.INTPRB) GO TO 21                                         HAC00600
      IF (J.GE.INTPRE) GO TO 21                                         HAC00610
C    PRINT CHECK OF FOURIER COEFFICIENTS OF CONVOLUTION.                HAC00620
      WRITE (4,301) XC(J),YC(J),J                                       HAC00630
  301 FORMAT(' ','                    ',2F10.5,I10)                     HAC00640
   21 CONTINUE                                                          HAC00650
      DO 22 J=1,N                                                       HAC00660
      T=2.*PY*(J-1)/FN                                                  HAC00670
      XD(J)=-SIN(T)*YC(J)/DR                                            HAC00680
      YD(J)=SIN(T)*XC(J)/DR                                             HAC00690
      IF (J.LE.INTPRB) GO TO 22                                         HAC00700
      IF (J.GE.INTPRE) GO TO 22                                         HAC00710
C  PRINT CHECK OF DIFFERENTIATION OF FOURIER COEFFICIENTS OF CONVOLUTIONHAC00720
      WRITE (4,401) XD(J),YD(J),J                                       HAC00730
  401 FORMAT (2F10.5,I10)                                               HAC00740
   22 CONTINUE                                                          HAC00750
      CALL CFFT2 (XD,YD,N,N,N,1)                                        HAC00760
      DO 30 K=1,N                                                       HAC00770
      WRITE (4,301) XD(K),YD(K),K                                       HAC00780
C    PRINT OUT OF VALUES OF INSIDE INTEGRAL.                            HAC00790
C    FIRST 'Q' POINTS ARE INVALID.                                      HAC00800
   30 CONTINUE                                                          HAC00810
      DO 35 I=257,321                                                   HAC00820
      WRITE (3,601) XD(I)                                               HAC00830
  601 FORMAT (F10.5)                                                    HAC00840
   35 CONTINUE                                                          HAC00850
      CALL CFFT2 (XC,YC,N,N,N,1)                                        HAC00860
C  OBTAINING THE INVERSE DFT OF FOUR. COEFF. OF CONVOLUTION.            HAC00870
      DO 40 L=1,N                                                       HAC00880
      IF (L.LE.INTPRB) GO TO 40                                         HAC00890
      IF (L.GE.INTPRE) GO TO 40                                         HAC00900
```

```
      WRITE (4,401) XC(L),YC(L),L                                       HAC00910
C   REAL COEFF. ARE IN FIRST COLUMN, IMAG. COEF. IN SECOND COL.         HAC00920
      4C CONTINUE                                                       HAC00930
      STCP                                                              HAC00940
      END                                                               HAC00950
```

APPENDIX F.3. EXPANDED FRINGE SHIFT INFORMATION USED IN JONES1 AND JONES2

This data was obtained from Van Houten's computer program for MODE ONE operation using control cards as explained in Van Houten's thesis [17] for the asymmetric Gaussian case with N = 65 and R = 1.5.

0.0
0.0
0.0
0.0
0.0
0.0
-0.11330
-0.24668
-0.22178
-0.15401
-0.06501
0.03319
0.13436
0.23487
0.33357
0.42932
0.51950
0.60518
0.68603
0.76181
0.83172
0.89571
0.95546
1.00798
1.05503
1.09827
1.13429
1.16478
1.18974
1.21074
1.22461
1.22293
1.23570
1.22293
1.22461
1.21074
1.18974
1.16478
1.13429
1.09827
1.05503
1.00798

```
 0.55546
 0.89571
 0.82172
 0.76181
 0.68603
 0.61518
 0.51550
 0.42932
 0.33457
 0.23487
 0.13436
 0.03319
-0.06501
-0.15401
-0.21213
-0.24668
-0.11330
 0.00000
 0.00000
 0.00000
 0.00000
```

APPENDIX G. COMPARISON OF JONES1 AND JONES2 VALUES OF THE INSIDE INTEGRAL WITH VAN HOUTEN'S VALUES OF THE INSIDE INTEGRAL

R	VALUE OF INSIDE INTEGRAL				% DIFFERENCE BETWEEN JONES1 AND VAN HOUTEN'S
	VAN HOUTEN'S	JONES1	JONES2		
-1.50000	0.69758	0.69181	0.69179	1	0.83
-1.45313	0.66847	0.67833	0.67839	2	-1.48
-1.40625	0.66746	0.60559	0.60561	3	9.27
-1.35938	0.37190	0.37536	0.37535	4	-0.93
-1.31250	-0.02912	-0.44074	-0.44074	5	-1413.53
-1.26563	-3.41609	-0.29548	-0.29548	6	91.35
-1.21875	2.84182	3.64357	3.64358	7	-28.21
-1.17188	12.47668	8.13610	8.13610	8	34.79
-1.12500	8.91488	8.48982	8.48980	9	4.77
-1.07813	6.96253	6.58171	6.58170	10	5.47
-1.03125	4.88435	4.85099	4.85099	11	0.68
-0.98438	3.38097	3.35730	3.35730	12	0.70
-0.93750	2.05348	2.10854	2.10849	13	-2.68
-0.89063	1.01299	1.06492	1.06489	14	-5.13
-0.84375	0.10356	0.16184	0.16183	15	-56.28
-0.79688	-0.73777	-0.61879	-0.61880	16	16.13
-0.75000	-1.34876	-1.26543	-1.26546	17	6.18
-0.70313	-1.90174	-1.81883	-1.81888	18	4.36
-0.65625	-2.40410	-2.31692	-2.31693	19	3.63
-0.60938	-2.86302	-2.75728	-2.75726	20	3.69
-0.56250	-3.23576	-3.12539	-3.12540	21	3.41
-0.51563	-3.49790	-3.44993	-3.44999	22	1.37
-0.46875	-3.89733	-3.74847	-3.74853	23	3.82
-0.42188	-4.06769	-3.97713	-3.97713	24	2.23
-0.37500	-4.21588	-4.17841	-4.17841	25	0.89
-0.32813	-4.53605	-4.38655	-4.38660	26	3.30
-0.28125	-4.62380	-4.54113	-4.54122	27	1.79
-0.23438	-4.74552	-4.64331	-4.64341	28	2.15
-0.18750	-4.78919	-4.74920	-4.74930	29	0.83
-0.14063	-5.00409	-4.85953	-4.85966	30	2.89
-0.09375	-5.00842	-4.92306	-4.92316	31	1.70
-0.04688	-5.04047	-4.94558	-4.94558	32	1.88
0.0	-5.04388	-4.95431	-4.95434	33	1.78
0.04688	-5.04047	-4.94553	-4.94565	34	1.88
0.09375	-5.00842	-4.92310	-4.92322	35	1.70
0.14063	-5.00409	-4.85958	-4.85969	36	2.89

0.18750	-4.78919	-4.74917	-4.74931	37	0.84
0.23438	-4.74552	-4.64322	-4.64340	38	2.16
0.28125	-4.62380	-4.54106	-4.54120	39	1.79
0.32813	-4.53605	-4.38656	-4.38657	40	3.30
0.37500	-4.21588	-4.17840	-4.17842	41	0.89
0.42188	-4.06769	-3.97706	-3.97714	42	2.23
0.46875	-3.89733	-3.74845	-3.74853	43	3.82
0.51563	-3.49790	-3.44993	-3.44999	44	1.37
0.56250	-3.23576	-3.12535	-3.12539	45	3.41
0.60938	-2.86302	-2.75719	-2.75725	46	3.70
0.65625	-2.40410	-2.31689	-2.31694	47	3.63
0.70313	-1.90174	-1.81887	-1.81887	48	4.36
0.75000	-1.34876	-1.26539	-1.26543	49	6.18
0.79688	-0.73777	-0.61872	-0.61878	50	16.14
0.84375	0.10356	0.16184	0.16182	51	-56.28
0.89063	1.01299	1.06493	1.06490	52	-5.13
0.93750	2.05348	2.10854	2.10852	53	-2.68
0.98438	3.38097	3.35732	3.35731	54	0.70
1.03125	4.88435	4.85099	4.85098	55	0.68
1.07813	6.96253	6.58171	6.58167	56	5.47
1.12500	8.91488	8.48975	8.48978	57	4.77
1.17188	12.47668	8.13600	8.13609	58	34.79
1.21875	2.84182	3.64350	3.64358	59	-28.21
1.26563	-3.41609	-0.29550	-0.29548	60	91.35
1.31250	-0.02912	-0.44073	-0.44074	61	-1413.50
1.35938	0.37190	0.37537	0.37534	62	-0.93
1.40625	0.66746	0.60559	0.60559	63	9.27
1.45313	0.66847	0.67835	0.67838	64	-1.48
1.50000	0.69758	0.69179	0.69181	65	0.83

REFERENCES

1. Gabor, D., "A New Microscopic Principle," Nature 161, 777, 1948.

2. Gabor, D., "Microscopy by Reconstructed Wavefronts," Proceedings of the Royal Society A197, 454, 1949.

3. Collier, R., Burckhardt, C.B., and Lin, L.H., Optical Holography, Academic Press, New York, 1971.

4. Collins, D., "Measurement of Velocities in Single and Two-Phase Flows," von Karmen Institute for Fluid Dynamics, Lecture Series 54, 19-23 February, 1973.

5. Radley, R.J., Jr. and Havener, A.G., "The Application of Dual Hologram Interferometry to Wind Tunnel Testing," AIAA Paper 73-210, Washington, D.C., 1973.

6. Chapman, D.R., Kuehn, D.M., and Larson, H.K., "Investigation of Separated Flows in Supersonic and Subsonic Streams with Emphasis on the Effect of Transition, "NACA TN 3869, 1957.

7. Korst, H.H., "A Theory for Base Pressures in Transonic and Supersonic Flow," ASME Journal of App. Mech. 23, 593-600, 1956.

8. Roache, P.J., "Numerical Solutions of Compressible and Incompressible Laminar Separated Flows," Ph.D. Thesis, University of Notre Dame, 1968, University Microfilms, Inc., Ann Arbor, Michigan, 1970.

9. Wazzan, A.R., "Review of Recent Developments in Turbulent Supersonic Base Flow," AIAA Journal 3, 1135, 1965.

10. Schlicting, H., Boundary Layer Theory, 6th ed., 26,41, McGraw-Hill Book Co., New York, N.Y., 1968.

11. Nassenstein, H., et al, Engineering Uses of Holography, 25, Cambridge University Press, 1970.

12. Sweeney, D.W., "Interferometric Measurement of Three-Dimensional Temperature Fields, "Ph.D. Thesis, University of Michigan, 1972.

13. Matulka, R.D., "The Application of Holographic Interferometry to the Determination of Asymmetric Three-Dimensional Density Field in Free Jet Flow," Ph.D. Thesis, Naval Postgraduate School, Monterey, Ca., 1970.

14. Rowley, P., "Quantitative Interpretation of Three-Dimensional Weakly Refractive Phase Objects Using Holographic Interferometry," Journal of the American Optical Society 59, 1496, 1969.

15. Junginger, H.G. and van Haeringer, W., "Calculation of Three-Dimensional Refractive Index Field using Phase Integrals," Optics Communications 5, 1, 1972.

16. Sweeney, D.W., "Interferometric Measurement of Three-Dimensional Temperature Fields," Ph.D. Thesis, University of Michigan, 1972.

17. Van Houten, P., "The Application of Holographic Interferometry to the Determination of Discontinuous Three-Dimensional Density Fields," A. E. Thesis, Naval Postgraduate School, Monterey, Ca., 1972.

18. Froberg, C., Introduction to Numerical Analysis, 195-223, Addison-Wesley Publishing Co., 1965.

19. Bergland, G.D., "A Guided Tour of the Fast Fourier Transform," Bell Telephone Laboratories, Inc., IEEE Spectrum, July, 1969.

20. Brigham, E.O., The Fast Fourier Transform, 198-221, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1974.

21. Everett, R., "Experimental Techniques in Holographic Interferometry," Naval Postgraduate School, Monterey, Ca., 1973.

22. Singleton, R.C., "An Algorithm for Computing the Mixed Radix Fast Fourier Transform," IEEE Transactions on Audio and Electroacoustics AU17-2, June, 1969.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library (Code 0212) Naval Postgraduate School Monterey, California 93940	2
3.	Chairman Department of Aeronautics Code 57 Naval Postgraduate School Monterey, California 93940	1
4.	Professor D.J. Collins Department of Aeronautics Code 57CO Naval Postgraduate School Monterey, California 93940	2
5.	LT Harland W. Jones, Jr., USN Air Department USS America CVA-66 FPO New York, New York	1

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)

Application of Holographic Interferometry to a Supersonic Flow Visualization of a Rear-Facing Step and Application of Fast Fourier Transform Methods to Current Data Reduction Techniques

5. TYPE OF REPORT & PERIOD COVERED
Master's Thesis; March 1974

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)

Harland Wilber Jones, Jr.

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Naval Postgraduate School
Monterey, California 93940

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Naval Postgraduate School
Monterey, California 93940

12. REPORT DATE
March 1974

13. NUMBER OF PAGES
90

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)
Naval Postgraduate School
Monterey, California 93940

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Holographic Interferometry
Fast Fourier Transform
Interferometric Measurement
Supersonic Flow Visualization
Rear-Facing Step Flow

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The successful application of holographic interferometry to the study of three-dimensional density fields around bodies in wind tunnel experiments has been reported in the literature along with the associated mathematical reduction processes for the basic interferometric equation.

The present report has extended the application of holography as a flow visualization technique by investigating Mach 2.8 flow over a rear-facing 1/8 in. step. Two single-exposed holograms of the model and two double-

DD FORM 1 JAN 73 1473 (Page 1)

EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. (cont'd)

exposed frozen fringe holographic interferograms of the flow were created.

Fast Fourier Transform (FFT) methods have been reported as significantly reducing computational time to obtain discrete and inverse discrete Fourier transforms. Two computer programs were designed to apply FFT methods to the Fourier transform approach to the basic interferometric equation. These programs were used to investigate an alternate way to calculate the value of the inside integral. Total agreement was not found, and further analysis as to the accuracy of each method is needed.